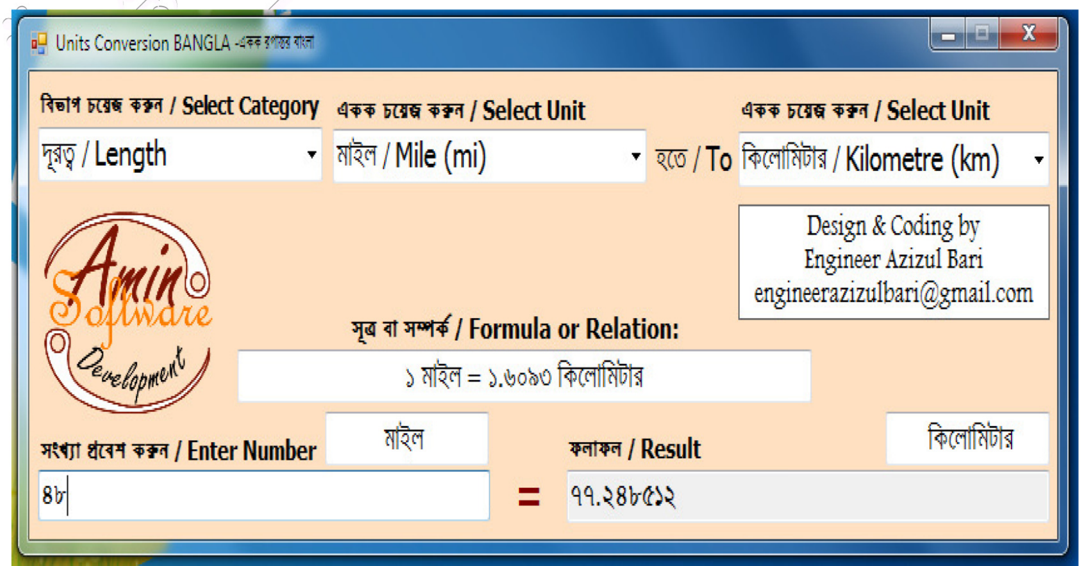
VB.NET

ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং

নতুন শিক্ষার্থীদের জন্য গুরুত্বপূর্ণ প্রশিক্ষণ রিসোর্স

প্রথম খণ্ড

Part ONE



বিশেষ ইন্টারনেট সংস্করণ

Special Internet Edition

ইঞ্জিনিয়ার আজিজুল বারী

engineerazizulbari@gmail.com

খানকায়ে আমীনিয়া-আসগরিয়া গবেষণা বিভাগ

Ali Centre, Subidbazar, Sylhet, BD

বিষয়সূচি

বিষয়সূচি

১ম পরিচ্ছেদ: *ভূমিকা*

মাইক্রসফ্‌ট ভিজুয়্যাল বেসিক.নেট - কী এটা?

এটা একটি সহজ কম্পিউটার প্রোগ্রামিং ভাষা। যে কেউ ইচ্ছে, এ ভাষা শিখে বিস্ময়কর সফ্‌টওয়্যার তৈরি করতে পারে পি.সি কম্পিউটারের জন্য। এ ভাষার মৌলিক ক'টি বৈশিষ্ট্য:

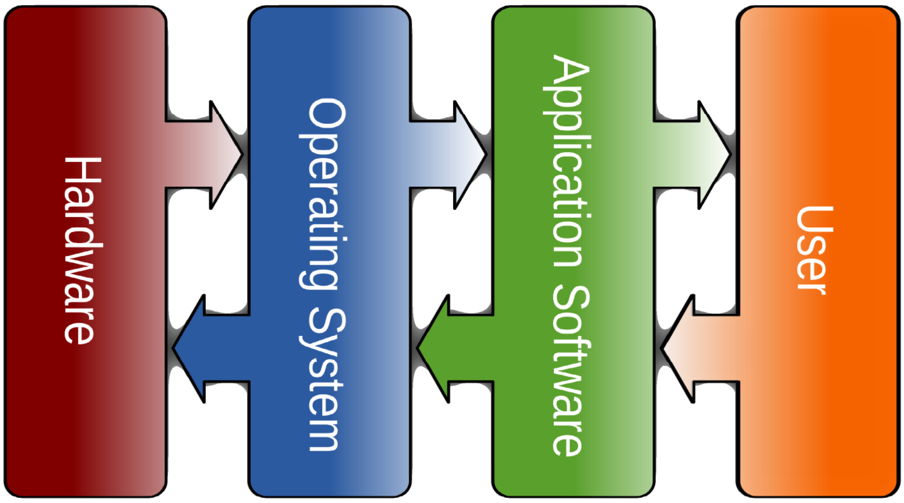
১. এটা বস্তুভিত্তিক প্রোগ্রামিং ভাষা (Object Oriented Programming Language)।
২. এটা আধুনিক, সহজ, জনপ্রিয়, ইংরেজির মতো ভাষা।
৩. এটা কাঠামোসর্বস্ব ভাষা (Structured Language)।
৪. এ ভাষায় লেখা প্রোগ্রামগুলো খুব নির্ভরশীল।
৫. ভুল-ত্রুটি সঠিককরণ (Debugging) অনেকটা সহজ।
৬. এটা মাইক্রসফটের ডট.নেট ফ্রেমওয়ার্ক (.Net Framework) এর অংশ।
৭. এ ভাষায় লিখিত সফ্‌টওয়্যার মাইক্রসফ্‌ট উইন্ডোজ এর জন্য খুব মানানসই।

জানেন কি?: পৃথিবীবিখ্যাত মাইক্রসফ্‌ট অফিস এর প্রতিটি এ্যাপলিকেশন ভিজুয়্যাল বেসিক (এবং ভিজুয়্যাল বেসিক.নেট) দ্বারা নিয়ন্ত্রণ ও রদবদল (customize) করা যায়।

এ পর্যন্ত যা বলা হলো তার এক বা একাধিক বিষয় আপনার বুঝে না আসলে ভাববেন না। এই গ্রন্থ ভালো করে অধ্যয়ন ও এতে সম্পৃক্ত বিভিন্ন পদে পদে বর্ণিত মডেল প্রোগ্রামগুলোর ওপর প্রাকটিস করে নিলে আপনিও হয়ে যাবেন ভিজুয়্যাল বেসিক.নেট প্রোগ্রামার। আপনাকে প্রোগ্রামার বানানোই আমাদের ইচ্ছে। পাঠ করতে থাকুন। দেখুন, কম্পিউটার প্রোগ্রামিং এর বিস্ময়কর জগৎ! এ বিস্ময়ের জগতে পাড়ি দিতে আপনার হাতিয়ার শুধুমাত্র একটি ল্যাপটপ বা ডেস্কটপ কম্পিউটার ও সেসাথে শেখার স্পৃহা- এই যা!

এই গ্রন্থে আপনি যা শিখবেন:

১. ভিজুয়্যাল বেসিক.নেট ভাষার মাধ্যমে সফ্‌টওয়্যার তৈরির পদ্ধতি।
২. উইন্ডোজের .exe এ্যাপলিকেশন তৈরি।
৩. উইন্ডোজের বেসিক ডাটাবেইজ প্রোগ্রামিং।



২য় পরিচ্ছেদ: শুরু করুন প্রাকটিস

কী লাগবে প্রাকটিস করতে?

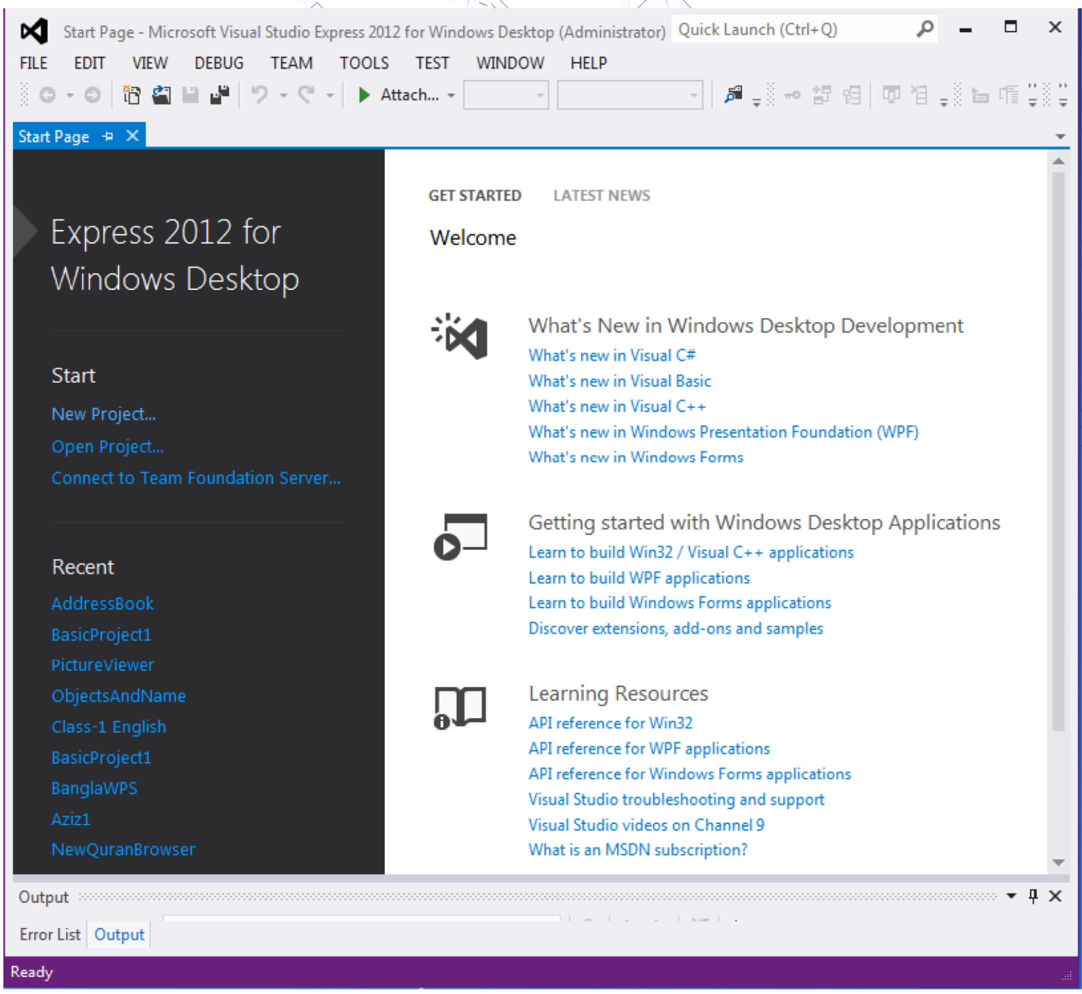
আই.ডি.ই (Integrated Development Environment)

বাংলায় বলা যায়, 'অঙ্গীভূত উন্নয়ন পরিবেশ'। মূলত আই.ডি.ই হলো সফ্টওয়্যার তৈরির একটি সফ্টওয়্যার। এতে আছে বিভিন্ন ধরনের হাতিয়ার (Tools) যা ব্যবহারের মাধ্যমে আপনি সফ্টওয়্যার সৃষ্টি করতে সক্ষম হবেন। আপনার সুবিধার্থে মাইক্রসফ্ট একাধিক উন্নতমানের আই.ডি.ই তৈরি করেছে। আপনার পছন্দসই আই.ডি.ই ইন্টারনেট থেকে (অথবা কম্পিউটারের দোকান থেকে) কম্পিউটারে ইনস্টল করতে পারেন। এজন্য আপনাকে তেমন বেশি টাকা দিতে হবে না। ইন্টারনেট থেকে ফ্রি ডাউনলোড করতে পারেন।

এই গ্রন্থে আমি মা্ইক্রসফ্টের **'ভিজুয়্যাল স্টুডিও এক্সপ্রেস ২০১২ ফর ডেস্কটপ'** থেকে উদাহরণ দিয়েছি। আপনি এটা ডাউনলোড করতে পারেন এই ওয়েভ ঠিকানা থেকে:

https://www.microsoft.com/en-us/download/details.aspx?id=34673

ইনস্টল করার পর মাইক্রসফ্টের ওয়েভসাইট থেকে বিনামূল্যে রেজিস্ট্রেশন করুন। এরপর ভিজুয়্যাল বেসিক এক্সপ্রেস ফর ডেস্কটপ যখনই রান করবেন তখন প্রথমে এই উইনডোটি আসবে:



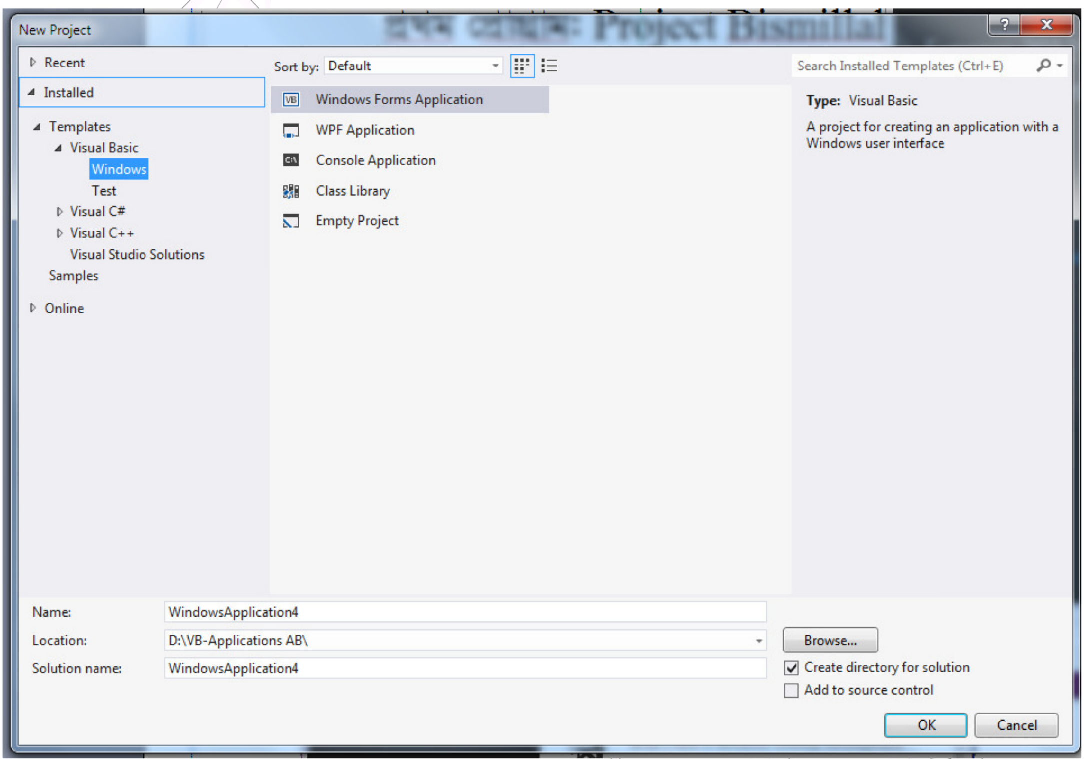
চিত্র-১: এক্সপ্রেস ২০১২ ফর উইন্ডোজ ডেস্কটপ রান করার পর প্রথমে প্রদর্শিত উইন্ডো।

১ম প্রোগ্রাম: ProjectBismillah

নিচে ধারাবাহিকভাবে বর্ণিত নির্দেশনাবলী অনুসরণ করে প্রজেক্ট তৈরি করুন:

১. Start এর নিচে লিখিত New Project... -এ ক্লিক করুন।

নতুন একটি প্রজেক্ট তৈরির জানালা আসবে। এটি নিম্নে দেখানো হলো।



চিত্র-২: New Project... -এ ক্লিক করার পর প্রদর্শিত জানালা।

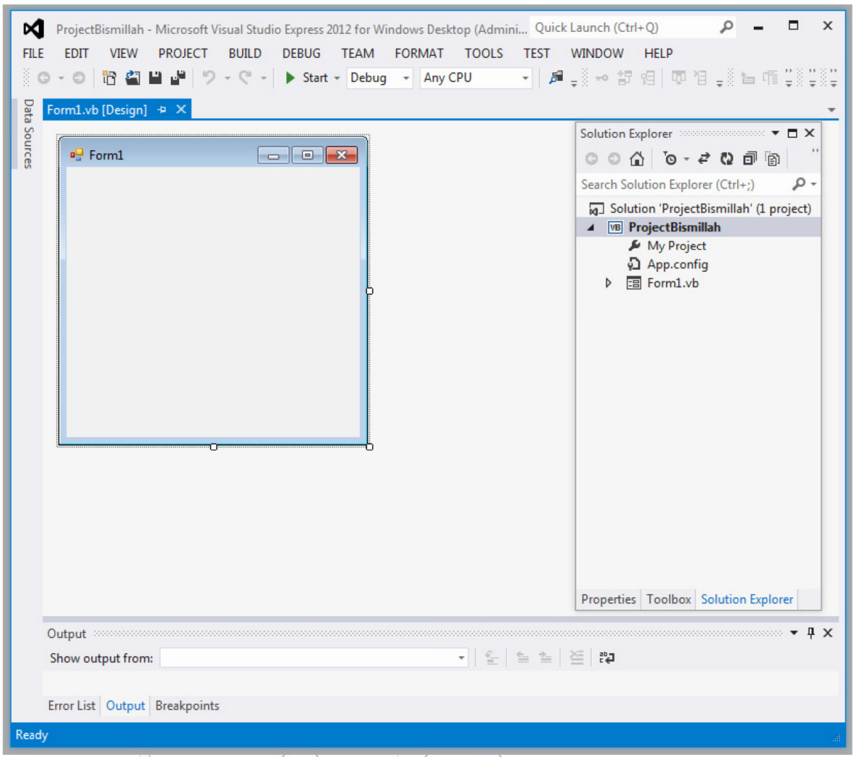
আপনি ডেস্কটপে LearnigDotNet নামকরণে একটি ফল্ডার বানিয়ে এর ভেতর সকল Project সংরক্ষণ করতে পারেন।

২. নতুন প্রজেক্ট ProjectBismillah -সংরক্ষণ করুন।

জানালার Name ফিল্ডে লিখুন ProjectBismillah। Windows Forms Application যেনো হাইলাইট (ক্লিক করা) থাকে। আমরা এই গ্রন্থে এরূপ প্রোগ্রামই করবো। Browse...-এ ক্লিক করে আপনার পছন্দসই ফল্ডারে সংরক্ষণের জন্য চয়েজ করুন। এরাব ক্লিক করুন 'Ok' বোতামে। আপনার নতুন প্রজেক্ট তৈরি হবে। কিছুক্ষণ পর প্রজেক্ট খুলবে এবং পরের পৃষ্ঠায় চিত্রিত জানালাটি প্রদর্শিত হবে।

ব্যস! আপনি প্রথম প্রজেক্ট তৈরি করে নিলেন। এবার আসুন, এই প্রজেক্টের উদ্দেশ্য হাসিলে কাজে লেগে যাই- কেমন?

১ম প্রোগ্রাম: ProjectBismillah



চিত্র-৩: ProjectBismillah তৈরির পর প্রদর্শিত আই.ডি.ই জানালা।

উপরের চিত্রের মতো আপনার কম্পিউটারে প্রদর্শিত আই.ডি.ই ঠিক হুবহু একই না-ও হতে পারে। কোনো ভাবনা নেই। যেভাবেই আসুক না কেনো, এতে সর্বদাই Form1 নামকরণে একটি ফরম থাকবে। এই ফরম বা জানালাটির ভেতরই আমাদের প্রোগ্রামের যাবতীয় বস্তু তৈরি হবে। একটু পরই আপনি বিষয়টি বুঝতে সক্ষম হবেন।

৩. সলিউশন এক্সপ্লোরার প্রদর্শন করুন।

যদি Solution Explorer জানালা প্রদর্শিত না হয়ে থাকে তাহলে VIEW ম্যানুতে ক্লিক করে চতুর্থ আইটেম Solution Explorer চয়েজ করুন। এটি টেনে ডানদিকে নিয়ে আসুন।

৪. টুলবক্স প্রদর্শন করুন।

যদি Toolbox জানালা প্রদর্শিত না হয়ে থাকে তাহলে VIEW ম্যানুতে ক্লিক করে ত্রয়োদশ আইটেম Toolbox চয়েজ করুন। এটি টেনে ডানদিকে নিয়ে আসুন। চাইলে Solution Explorer জানালার ভেতর ছেড়ে দিতে পারেন।

৫. প্রোপার্টিজ প্রদর্শন করুন।

যদি Properties জানালা প্রদর্শিত না হয়ে থাকে তাহলে VIEW ম্যানুতে ক্লিক করে শেষেরটির আগের আইটেম Properties Window চয়েজ করুন। এটি টেনে ডানদিকে নিয়ে আসুন। চাইলে Solution Explorer জানালার ভেতর এটাও ছেড়ে দিতে পারেন।

উপরে বর্ণিত উপায়ে কাজ করলে যে জানালাটি দেখা যাবে তা হবে এ পৃষ্ঠার চিত্রের মতো (চিত্র-৩)। লক্ষ করুন Solution Explorer জানালার নিচে তিনটি বোতাম আছে: Properties, Toolbox এবং Solution Explorer। যে সময় যেটির প্রয়োজন তাতে ক্লিক করলেই সে জানালা প্রদর্শিত হবে। আপাতত আমাদের দরকার এই তিনটি ও Form1 নামক জানালাটি।

ব্যস! ProjectBismillah -এর জন্য আপনি আই.ডি.ই প্রস্তুত করে নিয়েছেন। এবার কিছু প্রোগ্রামিং হবে।

১ম প্রোগ্রাম: ProjectBismillah

User Interface ডিজাইন

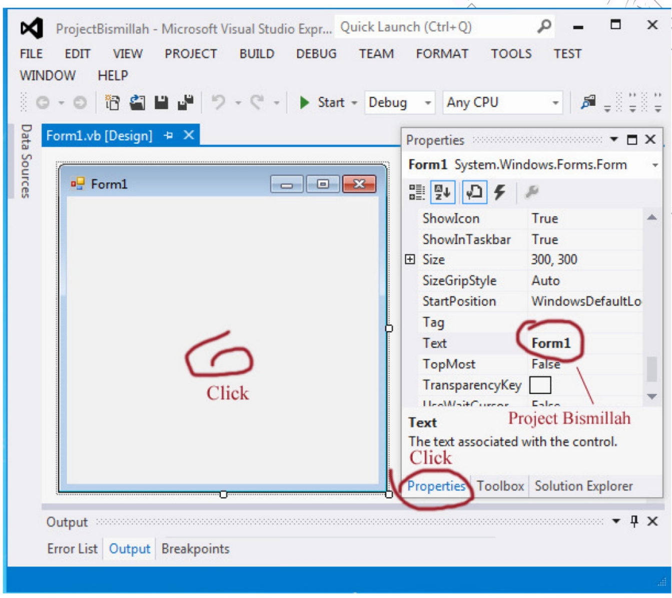
বাংলায় ব্যবহারকারীর সাথে প্রোগ্রামের মিথস্ক্রিয়াকে ইউজার ইন্টারফেইস বলা যায়। কিভাবে একজন ব্যবহারকারী আপনার প্রোগ্রাম থেকে কাঙ্ক্ষিত ফলাফল বা সুবিধা উপভোগ করবে তা সুন্দরভাবে ডিজাইন করার নামই হলো ইউজার ইন্টারফেইস। বিসমিল্লাহ প্রোগ্রামে আমরা যা চাচ্ছি তাহলো, একটি বোতামে ক্লিক করলে ইংরেজিতে 'Bismillahir Rahmanir Raheem' বাক্যটি একটা ফিল্ডে লেখা হবে। একই সময় একটি ডায়লোগ বা মেসেজ বাক্স আসবে ও "Beep" শব্দ হবে। বাক্সে ইংরেজিতে লেখা থাকবে "In the name of Allah the Merciful, the Beneficent." - এ কথাটি।

এ ইন্টারফেইস তৈরিতে আমাদের যা দরকার:

১. একটি প্রোগ্রাম ফরম বা উইন্ডো।
২. একটি লেখা প্রদর্শনের ফিল্ড বা টেক্সট বাক্স।
৩. একটি ক্লিক করার বোতাম।

উপরে উল্লেখিত তিনটি বস্তুর মধ্যে প্রথমটি তথা উইন্ডো ফরম ইতোমধ্যে আমরা তৈরি করে নিয়েছি। চাইলে একে সামান্য পরিবর্তন করতে পারি। যেমন, এর উপরের শিরোনাম '**Form1**' পরিবর্তন করে আমরা লিখতে পারি '**Project Bismillah**'। এটা করতে নিচের ধারাবাহিক পদক্ষেপগুলো গ্রহণ করুন।

১. ফর্মের ভেতর ক্লিক করে একে সিলেক্ট করুন। ২. সলিউশন এক্সপ্লোরারের নিচে 'Properties' শব্দের উপর ক্লিক করুন। ৩. স্ক্রল করে 'Text' আইটেমে যেয়ে 'Form1' শব্দ মুছে সেখানে লিখুন 'Project Bismillah'। এবার সেইভ করুন (Ctrl+Shift-S কিংবা File - Save All)। নিচের চিত্রটি (চিত্র-৪) দেখুন।



চিত্র-৪: Form1 -এর শিরোনাম পরিবর্তন।

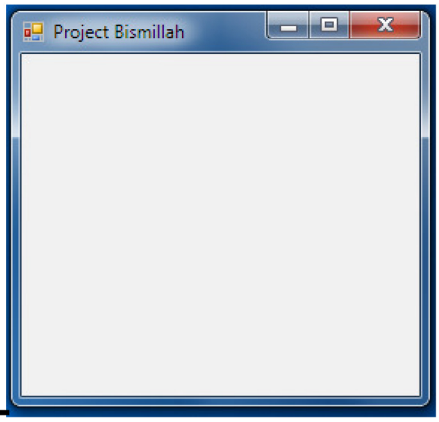
প্রোগ্রাম "Start" করা

যে কোনো প্রোগ্রাম তৈরির সময় বার বার এটি শুরু করে কাজ করছে কি না দেখতে হয়। এ সুবিধার জন্য "Start" নামকরণে একটি মেনু আইটেম দেওয়া আছে। এতে ক্লিক করলে প্রোগ্রাম বিল্ড করবে। প্রোগ্রামের কোডিংয়ে ভুল থাকলে সতর্ক মেসেজ প্রদর্শিত হবে। অন্যথায় সফলভাবে প্রোগ্রাম শুরু হবে। আপনি কাঙ্ক্ষিত ফলাফল পাচ্ছেন কি না পরীক্ষা করে নিতে পারেন। এরপর এটা বন্ধ করলেই আই.ডি.ই -তে ফিরে আসবেন।

এ পর্যন্ত আমরা যা করেছি তা দেখার জন্য একবার "Start" বোতামে ক্লিক করুন। নিচের চিত্র-৫ এর মতো একটি উইন্ডো দেখা যাবে।

মনে রাখবেন:

কিছুক্ষণ কাজ করেই **Ctrl+Shift-S** কিংবা **File - Save All** দ্বারা প্রজেক্ট সেইভ বা সংরক্ষণ করবেন। এভাবে সংরক্ষণের অভ্যেস করে নেবেন। আপনার কাজ অনাকাঙ্ক্ষিত কোনো কারণে নষ্ট হবে, তা-তো মানা যায় না!



চিত্র-৫: Start -করার পর প্রদর্শিত জানালা

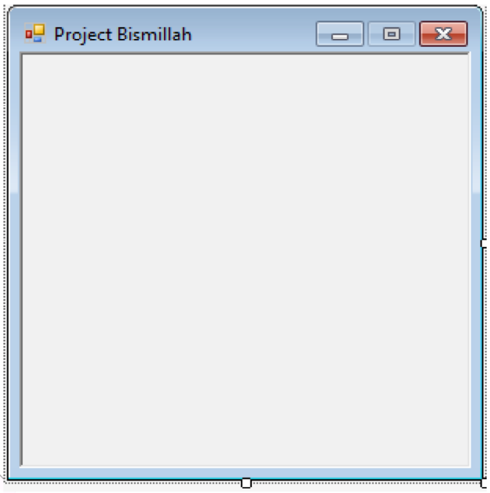
১ম প্রোগ্রাম: ProjectBismillah

FormBorderStyle ও MaximumBox ডিজাইনে পরিবর্তন

প্রোগ্রাম চলার সময় যাতে ফরম জানালার আয়তন পরিবর্তন না হয় সে জন্য দু'টি প্রোপার্টিতে পরিবর্তন করলে ভালো হবে। এই দু'টি পরিবর্তন করতে নিম্নের ধারাবাহিক পদক্ষেপগুলো অনুসরণ করুন:

ফর্মে ক্লিক করে এটাকে সিলেক্ট করুন। ক্লিক করুন সলিউশন এক্সপ্লোরার জানালায়। এবার স্ক্রল করে FormBorderStyle -এ যান। ডানের ড্রপডা্উন মেনু থেকে চয়েজ করুন Fixed3D আইটেম। এরপর স্ক্রল ক ে র চলে যান MaximumBox -এ। এর ডানের বাক্সে True লেখা থকালে ড্রপডাউন বাক্স থেকে False শব্দটি চয়েজ করুন।

এবার আপনার ফর্মের চেহারা ডানের চিত্রের মতো হবে। প্রোগ্রাম Start করে দেখুন- ফর্মের সাইজ পরিবর্তন করা আর সম্ভব হবে না।



চিত্র-৬: Form1 -এর চিত্র।

একটি Button ও একটি Text box ফর্মে যুক্তকরণ

আমরা ফর্মের ডিজাইন শেষ করেছি। এবার এর ভেতর দু'টি বস্তু যুক্ত করবো। প্রথমে একটি Text box ফর্মের উপরে এনে বসাবো। নিচে বর্ণিত পদক্ষেপগুলো ধারাবাহিকভাবে অনুসরণ করুন।

ক্লিক করুন সলিউশন এক্সপ্লোরার জানালায়। ক্লিক করুন নিচের Toolbox বাক্সে। Common Controls -এর নিচে দেখতে পাবেন অনেকগুলো আইটেম। ডবল ক্লিক করুন ২নং আইটেম Button-এ। দেখবেন, Button1 -নামক একটি বোতাম ফর্মের উপরের বায়ের কোণে চলে গেছে। আপনি বোতাম যোগ করে নিয়েছেন।

এবার টুল বাক্সে যেয়ে স্ক্রল করে নিচে যান। আগের মতো ডবল ক্লিক করুন TextBox -নামক আইটেমে। লক্ষ করুন, একটি বাক্স চলে গেছে ফর্মের ভেতর। এটাকে মাউস দিয়ে টেনে মাঝখানে নিয়ে আসুন। বোতামটিও নিচে সেন্টারে নিয়ে আসুন।

টেক্সট্ বাক্স ও বোতামের প্রোপার্টি পরিবর্তন

আমরা চাই টেক্সট্ বাক্সের লেখাটুকু একটু বড় সাইজ হবে। এছাড়া বোতামে লেখা থাকবে Click Me শব্দগুলো। লেখার সাইজও কিছুটা বড় হবে। আমরা এ কাজগুলো সহজে করতে পারি প্রোপার্টিজ জানালা ব্যবহার করে। নিম্নে বর্ণিত পদক্ষেপগুলো অনুসরণ করুন।

Button - ক্লিক করে Button1 সিলেক্ট করুন। এক্সপ্লোরার জানালার নিচে Properties বাক্সে ক্লিক করুন। এবার প্রথম দিকের + Font -এর + চিহ্নে ক্লিক করুন। আরো একটি মেনু আসবে। এখানে Size কথাটির বাক্সে ৮ মুছে ১৪ লিখুন। প্রোপর্টিজ জানালার ডানের স্ক্রলে ধরে নিচের দিকে যেয়ে Text আইটেম বের করুন। এর ডানে Button1 শব্দটি মুছে Click Me শব্দদ্বয় লিখুন। ব্যস! বোমাতের পরিবর্তন সম্পূর্ণ হলো।

TextBox - ক্লিক করে টেক্সট্ বাক্স সিলেক্ট করুন। প্রোপার্টিজ জানালায় যেয়ে আগের মতো ফন্ট সাইজ বদলে ১৪ করুন। ব্যস! হয়ে গেল।

ফর্মের সাইজ পরিবর্তন, টেক্সট্ বাক্স ও বেতামের অবস্থান পরিবর্তন শেষে যে ফর্ম তৈরি হলো তা নিম্নে দেখানো হয়েছে।



চিত্র-৭: Form1 -এর শেষ চিত্র।

১ম প্রোগ্রাম: ProjectBismillah

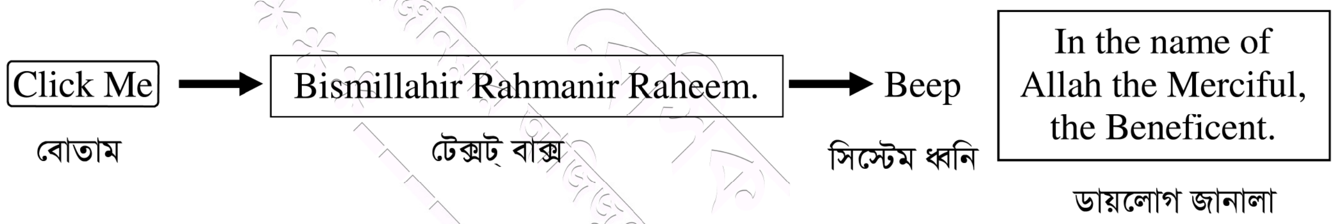
Designer থেকে Programmer

এ পর্যন্ত আপনি যা করলেন তাতে আপনাকে প্রোগ্রামার উপাধিতে ভূষিত করা যাবে না! হ্যাঁ, আপনি ডিজাইনার ছিলেন। কারণ আপনি একটি ইউজার ইন্টারফেইস ডিজাইন করেছেন সফলভাবে। তবে আমাদের ইচ্ছা আপনি শুধু ডিজাইনার নয় একজন দক্ষ প্রোগ্রামার হবেন। অর্থাৎ বুঝেসুঝে আপনি প্রোগ্রাম চলাকালের এ্যাকশন কোড লিখার যোগ্যতা লাভ করবেন। আর এটাই তো এ গ্রন্থ রচনার উদ্দেশ্য।

প্রোগ্রাম করা যেমন কঠিন মনে করা হয় বাস্তবে তা কিন্তু নয়। অবশ্য এজন্য মাইক্রসফটকে ধন্যবাদ দিতেই হবে। এক সময় এমন ছিলো, আপনি যে ইউজার ইন্টারফেইস কয়েক মিনিট বা ধরুণ ঘণ্টা খানেকের মধ্যেই তৈরি করে নিলেন তা করতে, লিখতে হতো শত শত লাইনসর্বস্ব কোড। সময় লাগতো কয়েক দিন পর্যন্ত। কিন্তু এ কাজগুলো আমরা রেডি-মেইড পেয়ে যাচ্ছি। যা তারা রেডি-মেইড করে দিতে পারেন নি তাহলো আমাদের চাহিদা। কারণ তারা তো জানেন না, প্রোগ্রাম দ্বারা আমরা কি করতে চাই? এ কারণেই বাকী অংশ নিজেরাই করতে হবে।

প্রোগ্রাম কোডিং

আমরা চাই, যে মুহূর্তে ব্যবহারকারী Click Me বোতামে ক্লিক করবেন, তখনই কিছু তথ্য প্রদর্শন করতে। কিসব তথ্য প্রদর্শিত হবে তা নিম্নে চিত্রিত হয়েছে।



উইন্ডোজ ফরম এপলিকেশনের কোডিং ফিল্ডে সর্বদাই নিম্নে চিত্রিত একটি কোড ব্লক থাকে। এই ব্লকের ভেতরেই যাবতীয় কোড লিখতে হবে।

```
Public Class Form1

   সব কোড এখানে

End Class
```

Click Me - বোতামের কোড

ডিজাইন উইন্ডোতে চিত্রিত বোতামে ডবল ক্লিক করলেই নিচের ফিল্ডটি আসবে। আমাদের লেখা লাইনটি আন্ডারলাইন করে দেখানো হয়েছে। মূল কোডে আন্ডারলাইন থাকবে না।

```
Public Class Form1

   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
         TextBox1.Text = "Bismillahir Rahmanir Raheem."
   End Sub

End Class
```

এই `TextBox1.Text = "Bismillahir Rahmanir Raheem."` লাইন কোড দ্বারা আমরা নির্দেশ দিচ্ছি টেক্সট বাক্সে যা লিখা হবে।

আমরা এবার Click Me বেতামের Sub-এ আরো দু'টি লাইন যুক্ত করে নেবো। লাইন দুটো হলো:

```
Beep()
MsgBox("In the name of Allah the Merciful, the Beneficent.")
```

ব্যস! কোডিং শেষ হলো। Form1.vb* -নামক ফিল্ডে এবার যা লিখা থাকবে তাহলো:

```
Public Class Form1
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      TextBox1.Text = "Bismillahir Rahmanir Raheem." 'what we want to be written in Text Box.
      Beep() 'System sound.
      MsgBox("In the name of Allah the Merciful, the Beneficent.") 'The text for message box window.
   End Sub
End Class
```

ব্যাখ্যা: Click Me বোতামের Click নামক একটি Event আছে। আমরা এই ইভেন্টটি ধরে কম্পিউটারকে ৩টি কোড লাইনে নির্দেশ দিচ্ছি। যথা: লাইন-১ দ্বারা বলা হচ্ছে ফর্মের TextBox1-এ কী লেখা হবে। লাইন-২ দ্বারা নির্দেশ দিচ্ছি কম্পিউটারের Beep নামক ধ্বনিটি প্লে করার জন্য। এটি সিস্টেমে সংরক্ষিত আছে। লাইন-৩ দ্বারা বলা হচ্ছে একটি মেসেজ বাক্স খুলে তাতে কিছু লিখে প্রদর্শন করতে।

দ্র: লক্ষ করুন, কোডে সবুজ কিছু লেখা আছে। এগুলোর শুরুতে ' -এ সিঙ্গুল কোট মার্কও দেওয়া হয়েছে। এ সবুজ লাইনগুলো হলো 'কমেন্ট'। ভিজুয়্যাল বেসিক এগুলো সম্পূর্ণরূপে উপেক্ষা করে যাবে। প্রোগ্রামারের সুবিধার্তে এই কমেন্ট লাইনের ব্যবস্থা। মন্তব্য করলে কোডের অর্থ অনেকটা স্পষ্ট হয়ে ওঠে। আপনিও অবশ্যই মন্তব্য করবেন। এগুলো রেডি-হওয়া প্রোগ্রামের কোনো ক্ষতি করে না কিংবা এদের কারণে অতিরিক্ত মেমোরিও খরচ হয় না। তবে প্রয়োজনে প্রোগ্রাম সঠিককরণের [ডিবাগিং] সময় এসব মন্তব্য বিরাট উপকারী হবে। মনে রাখবেন, কোনো কোনো প্রোগ্রামের কোডের লাইন সংখ্যা শত শত থেকে হাজার হাজারে যেয়ে পৌঁছতে পারে!

১ম প্রোগ্রাম শেষ!

আপনি সফলভাবে প্রথম প্রোগ্রাম শেষ করেছেন। তবে আই.ডি.ই বন্ধ করার পূর্বে অবশ্যই একবার প্রোগ্রাম টেস্ট করে দেখতে হবে। নিচের চিত্র-৮-এ এই টেস্টের ফলাফল চিত্রিত হলো।



চিত্র-৮: ১ম প্রোগ্রাম ProjectBismillah পরিচালনা করে Click Me বোতামে ক্লিক করার ফলাফল।

৩য় পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার মৌলিক কাঠামো*

ভিজুয়্যাল বেসিক.নেট : Subs & Functions

ডট.নেট প্রোগ্রামের কোড মূলত Sub [Subroutine] ও Function নিয়ে গঠিত। ১ম প্রোগ্রামে আমরা Sub ব্যবহার করেছি। এ দু'টো হলো প্রোগ্রাম কোডের মধ্যে একেকটি অংশ।

প্রত্যেক Sub শুরু হয় এভাবে [এর Syntax] :

```
[accessibility] Sub routine_name [paramaters]
    statements
End Sub
```

accessibility = Public, Protected, Friend, Protected Friend, or Private

Public -অর্থ এই সাবরুটিন সমগ্র প্রোগ্রামব্যাপী ব্যবহারযোগ্য।

Private -অর্থ যে মডিউলের ভেতর এটি আছে শুধু সেখানেই ব্যবহারযোগ্য।

আপাতত এ দু'টোই আমরা ব্যবহার করবো।

routine_name -যে কোনো নাম। নামটি ভিজুয়্যাল বেসিকের আইন মুতাবিক হতে হবে। যে কোনো অক্ষর বা আন্ডারস্কোর [_] দ্বারা নাম শুরু হবে। নামের মধ্যে কোনো স্পেইস [শূন্যস্থান ক্যারাক্টার] থাকবে না।

paramaters - ঠিক ভেরিয়েবল্ ঘোষণার মতো। একটু পর কিছু উদাহরণ দ্বারা সব পরিষ্কার হবে।

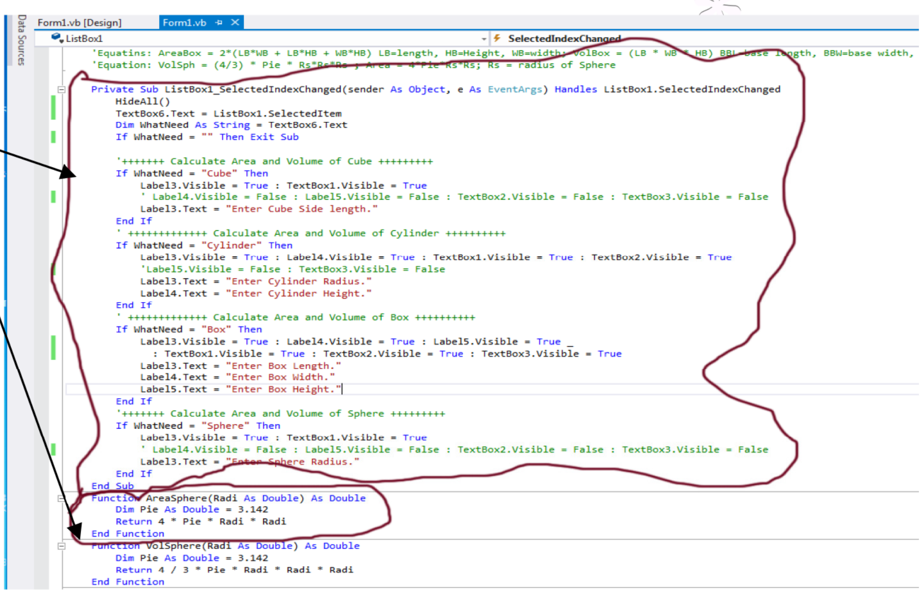
প্রত্যেক Function শুরু হয় এভাবে [এর Syntax]:

```
[accessibility] Function function_name [paramaters] [As return_type ]
    statements
End Function
```

প্রতিটি আইটেম Sub ঘোষণার অনুরূপ। আর As return_type -এর অর্থ হলো কোন্ ধরনের তথ্য ফাংশন ফিরে দেবে।

মনে রাখবেন: প্রত্যেক ফাংশনের কাজ হলো কোনো তথ্য [ভ্যালু] ফিরে দেওয়া।

বিষয়টি স্পষ্টভাবে বুঝার জন্য নিচের কোড অংশ দেখুন (চিত্র-৯)। এই কোডিংয়ের উদাহরণ পরে আসছে।



```
'Equatins: AreaBox = 2*(LB*WB + LB*HB + WB*HB) LB=length, HB=Height, WB=width; VolBox = (LB * WB * HB) BBL=base length, BBW=base width,
'Equation: VolSph = (4/3) * Pie * Rs*Rs*Rs ; Area = 4*Pie*Rs*Rs; Rs = radius of Sphere

Private Sub ListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ListBox1.SelectedIndexChanged
    HideAll()
    TextBox6.Text = ListBox1.SelectedItem
    Dim WhatNeed As String = TextBox6.Text
    If WhatNeed = "" Then Exit Sub

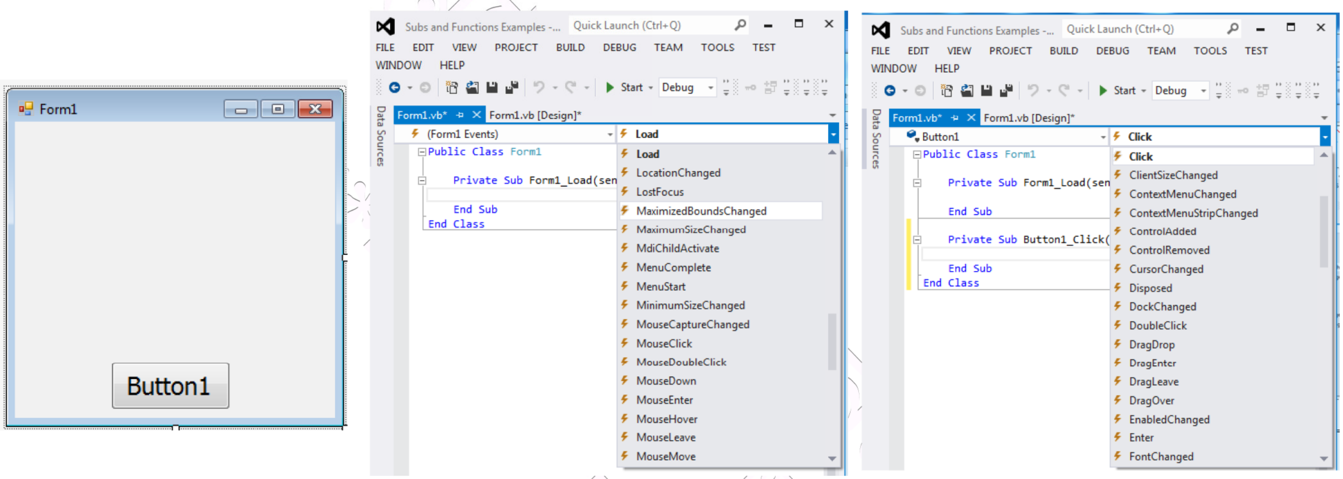
    '+++++++ Calculate Area and Volume of Cube +++++++++
    If WhatNeed = "Cube" Then
        Label3.Visible = True : TextBox1.Visible = True
        ' Label4.Visible = False : Label5.Visible = False : TextBox2.Visible = False : TextBox3.Visible = False
        Label3.Text = "Enter Cube Side length."
    End If
    ' +++++++++++++ Calculate Area and Volume of Cylinder ++++++++++
    If WhatNeed = "Cylinder" Then
        Label3.Visible = True : Label4.Visible = True : TextBox1.Visible = True : TextBox2.Visible = True
        'Label5.Visible = False : TextBox3.Visible = False
        Label3.Text = "Enter Cylinder Radius."
        Label4.Text = "Enter Cylinder Height."
    End If
    ' +++++++++++++ Calculate Area and Volume of Box ++++++++++
    If WhatNeed = "Box" Then
        Label3.Visible = True : Label4.Visible = True : Label5.Visible = True _
          : TextBox1.Visible = True : TextBox2.Visible = True : TextBox3.Visible = True
        Label3.Text = "Enter Box Length."
        Label4.Text = "Enter Box Width."
        Label5.Text = "Enter Box Height."
    End If
    '+++++++ Calculate Area and Volume of Sphere +++++++++
    If WhatNeed = "Sphere" Then
        Label3.Visible = True : TextBox1.Visible = True
        ' Label4.Visible = False : Label5.Visible = False : TextBox2.Visible = False : TextBox3.Visible = False
        Label3.Text = "Enter Sphere Radius."
    End If
End Sub
Function AreaSphere(Radi As Double) As Double
    Dim Pie As Double = 3.142
    Return 4 * Pie * Radi * Radi
End Function
Function VolSphere(Radi As Double) As Double
    Dim Pie As Double = 3.142
    Return 4 / 3 * Pie * Radi * Radi * Radi
End Function
```

চিত্র-৯: Sub এবং Function লেখা হয়েছে ফর্মের ক্লাস ব্লকের ভেতর।

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

১ম উদাহরণ: Sub

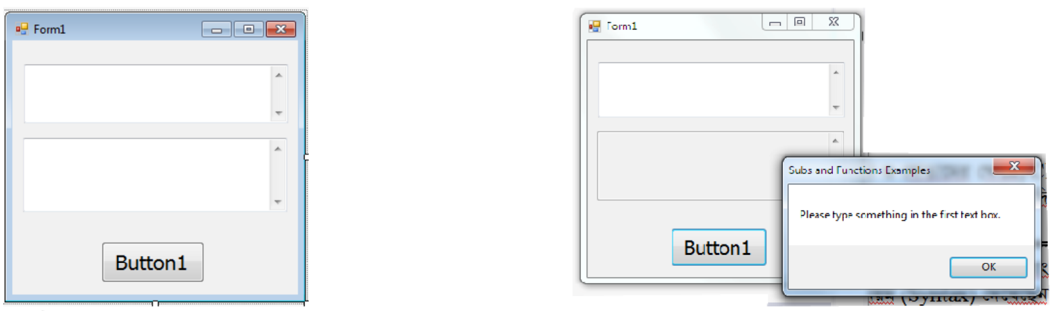
Sub এর উদাহরণ উপস্থাপনের পূর্বে আপনাকে একটি বিষয় খুব ভালো করে জানতে হবে। Form, Button, TextBox, Label, ListBox, ComboBox ইত্যাদি টুলবাক্সের যাবতীয় বস্তুর [Objects] একাধিক event [ঘটনা] আছে। প্রোগ্রামার হিসাবে আপনার কাজ হলো এসব ঘটনাকে Sub -এর মধ্যে ধরে নিজের ইচ্ছে মতো কাজ করানো। নিচের চিত্রে [চিত্র-১০] ফরম ও বোতামের কিছু ঘটনা [events] দেখানো হয়েছে। এসব ইভেন্ট কোড উইন্ডোতে যেয়ে সিলেক্ট করলেই এককেটি Sub আই.ডি.ই বানিয়ে দেবে। আপনি এর ভেতর কোড লিখে ইভেন্টকে নিয়ন্ত্রণ ও কাজে লাগাবেন। আর এভাবে ইভেন্ট কাজে লাগানোর নামই প্রোগ্রামিং। আপনি যদি সফলভাবে এগুলো কাজে লাগাতে সক্ষম হন তাহলে আপনি হবেন একজন দক্ষ প্রোগ্রামার।



চিত্র-১০: (বায়ে) একটি Project। (মাঝে) প্রজেক্টের ফর্মের ইভেন্ট লিস্ট। (ডানে) প্রজেক্টের বাটন১ এর ইভেন্ট লিস্ট।

অধিকাংশ ক্ষেত্রে আপনি যেসব ইভেন্ট কাজে লাগাবেন তাহলো: (ফরম ও বোতামের ক্ষেত্রে) Click, MouseUp, MouseDown, MouseEnter, MouseLeave; (লিস্টবাক্স, কম্বোবাক্স, টেক্সটবাক্স ইত্যাদির ক্ষেত্রে) Click, MouseEnter, TextChanged, MouseLeave ইত্যাদি।

Sub শব্দের অর্থ Subroutine। ভিজুয়্যাল বেসিক.নেট -এ সাব -এর গুরুত্ব অপরিসীম। সুতরাং সাব কিভাবে কাজ করে তা আপনাকে জানতেই হবে। আপনি আগের পৃষ্ঠায় সাব লেখার নিয়ম (Syntax) দেখেছেন। পরের পৃষ্ঠার সাব দু'টি উপরের চিত্র-১০ -এ চিত্রিত বোতামের Click event ও টেক্সট বাক্সের MouseLeave event সাব, যা আমরা লিখেছি দু'টি কাজ আঞ্জাম দিতে। উক্ত ফর্মে এবার দু'টি TextBox কন্টোলও যোগ করেছি। নিচের বায়ের চিত্রে (চিত্র-১১) দেখানো হয়েছে ফর্মের ডিজাইনটি। ডানের চিত্রে (চিত্র-১২) তুলে ধরেছি রান করার পরে বোতামে ক্লিক করার ফলাফল। আপনি যখন মেসেজ বাক্সের 'Ok' বোতামে ক্লিক করবেন তখনই ফিল্ডে কিছু লিখার জন্য প্রোগ্রাম অপেক্ষায় থাকবে।



চিত্র-১১: (বায়ে) ফর্মের ডিজাইন। (ডানে) প্রোগ্রাম চলাকালে বোতামে ক্লিক করলে যে মেসেজ আসবে তার দৃশ্য।

৩য় পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার মৌলিক কাঠামো*

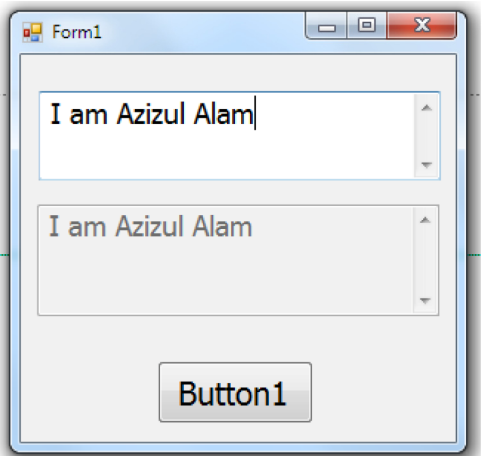
ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

আগের পৃষ্ঠায় বর্ণিত উদাহরণের দু'টি Sub নিচে তুলে ধরেছি। এরপর (চিত্র-১৩) প্রোগ্রাম চলাকালে প্রথম টেক্সট্ বাক্সের লেখা বাক্স থেকে মাউস সরে নিয়ে আসার পর দ্বিতীয় টেক্সট্ বাক্সে লেখা কপি হয়েছে দেখা যাচ্ছে।

```
Public Class Form1
   'This sub is the button click event code
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      MsgBox("Please type something in the first text box.") 'call messageBox to display
      'instruction
   End Sub

   'Catching MouseLeave event and doing something
   Private Sub TextBox1_MouseLeave(sender As Object, e As EventArgs) Handles TextBox1.MouseLeave
      If TextBox1.Text = "" Then Exit Sub 'checks if there is text in the first field. if not program ends.
      TextBox2.Text = TextBox1.Text  'if there is text, this line executed, copies
      ' text into 2nd field.
   End Sub
End Class
```

উক্ত সাব দু'টো কিভাবে কাজ করে: প্রোগ্রাম রান করলেই পুরো ফর্মটি প্রদর্শিত হবে। আপনি যদি বোতামে ক্লিক করেন তাহলে প্রথম সাবটি কাজ শুরু করবে। এই সাবে আমরা নির্দেশ দিয়েছি: একটি মেসেজ বাক্স প্রদর্শন করতে যেখানে লিখা থাকবে, **"Please type something in the first text box."** কথাটি। **'Ok'** ক্লিক করার পর আগের অবস্থায় ফিরে যাবে। এবার আপনি যদি কিছু লিখেন এবং মাউসটি বাক্স থেকে বের করেন তাহলে সাথে সাথে লেখাটুকু এমনিতেই দ্বিতীয় বাক্সে চলে যাবে। এই কাজটি সম্ভব হয় ১ম টেক্সট্ বাক্সের মাউসলীভ সাব সক্রিয় হওয়ার কারণে। এ সাবে আমরা নির্দেশ দিয়েছি: প্রথমে পরীক্ষা করো, বাক্সে কিছু লেখা হলো কি না? যদি না হয়- সাব থেকে বের হও। যদি লেখা থাকে তাহলে দ্বিতীয় লাইনের নির্দেশ পালন করো। অর্থাৎ ১ম বাক্সের যাবতীয় লেখা ২য় বাক্সে লিখ। এরপর সাব থেকে বের হও।



চিত্র-১৩: প্রোগ্রাম Run করে প্রথম টেক্সট্ বাক্সের লিখা স্বয়ংক্রিয়ভাবে দ্বিতীয়টি লেখা হওয়ার দৃশ্য।

২য় উদাহরণ: Function

উপরের ১ম উদাহরণে আমরা Sub-এর ব্যবহার দেখেছি। এবার একটি নিজের তৈরি ফাংশন (Function) উপস্থাপন করছি। ভিজুয়্যাল বেসিকে Function সর্বদাই কোনো এক বা একাধিক ভ্যালু (value) রিটার্ন করে। এটা হতে পারে প্রাইভেট বা পাবলিক বা আরো কোনো ধরনের। ইতোমধ্যে আপনি ফাংশনের সিন্টেক্স (Syntax) দেখেছেন। যাক, উক্ত প্রজেক্টে এখন আমরা একটি ফাংশন যোগ করবো।

** **মনে রাখবেন: ফাংশন ব্লক ফর্মের ক্লাসের ভেতর যে কোনো স্থানে লিখা যায়। এটাকে যে কোনো সাব বা অপর ফাংশন থেকে কল (call) করা হয়ে থাকে। ফাংশন (এবং সাবকে) বার বার কল করা সম্ভব। আর সর্বদাই ফাংশন থেকে ফলাফল (ভ্যালু) আসবে। ভ্যালুর ধরন (type) যে কোনোটি হতে পারে, যেমন: text, number, etc.** **

উদ্দেশ্য: আমাদের ফাংশনের কাজ হলো, আপনি মোট ক'টি অক্ষর টাইপ করেছেন তা একটি মেসেজ বাক্সে দেখানো। সুতরাং ফাংশনের রিটার্ন ভ্যালু হবে একটি ইন্টেজার (আস্ত) সংখ্যা। পরের পৃষ্ঠায় এ সম্পর্কে বিস্তারিত দেখুন।

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

আমাদের ফর্ম ডিজাইনে কোনো পরিবর্তনের প্রয়োজন নেই। আমাদের কাজ হলো ফাংশন ব্লকটি ফর্মের ক্লাসের ভেতর লেখা। এরপর এটি অন্য কোনো কোড ব্লক থেকে কল করা। এই যা। নিচে ফাংশন কোডসহ পুরো কোড তুলে ধরেছি।

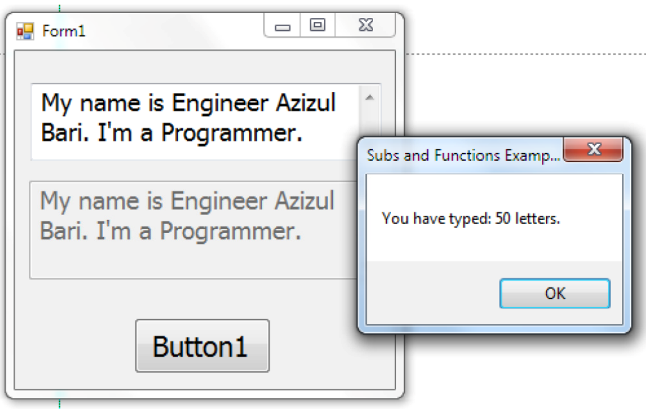
```
Public Class Form1
    'This sub is the button click event code
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        MsgBox("Please type something in the first text box.") 'call messageBox to display
        'instruction
    End Sub
    'Caching MouseLeave evet and doing something
    Private Sub TextBox1_MouseLeave(sender As Object, e As EventArgs) Handles TextBox1.MouseLeave
        If TextBox1.Text = "" Then Exit Sub 'chacks if there is text in the first field
        TextBox2.Text = TextBox1.Text   'if there is text this line executed. copies
        ' text into 2nd field.
        'function call from message box
        MsgBox("You have typed: " & NumOfChar(TextBox1.Text) & " letters.")
    End Sub
    'function returns number of characters in TextBox1
    Function NumOfChar(Supply As String) As Integer
        If TextBox1.Text = "" Then 'in case there aren't any characters in the textBox1
     'although it is checked at the beginning of the sub. This If --- End If block is for extra solidarity.
            Return 0 'empty textBox means no character.
        End If
        Return Len(TextBox1.Text) 'counts character and returns integer
    End Function
End Class
```

প্রোগ্রামে ফাংশনের ব্যবহার: আমাদের সৃষ্ট NumOfChar (Supply) ফাংশন কিভাবে ব্যবহৃত হয়েছে, তা একটু লক্ষ্য করুন। টেক্সটবক্স১ এর 'TextBox1_MouseLeave' ইভেন্ট সাব থেকে এটি কল করেছি। ফলাফল একটি সংখ্যা যা মেসেজ বাক্সে লিখা হবে। অন্যকথায় মেসেজ বাক্সের ভেতর থেকেই আমরা কলটি করেছি। চাইলে এভাবেও হতো:

```
Dim NumOfLetters as Integer
NumOfLetters = NumOfChar(TextBox1.Text) 'Function call
MsgBox("You have typed: " & NumOfLetters & " letters.")
```

উক্ত কোড দ্বারা প্রথমে আমরা NumOfLetters নামক একটি ভেরিয়েবলে বাক্সের অক্ষর সংখ্যা ফাংশন কল করে সংরক্ষণ করেছি। এরপর এটা মেসেজ বাক্স দ্বারা প্রদর্শিত হয়েছে। তবে এ ক্ষেত্রে দু'টি অতিরিক্ত লাইন কোডিং হয়েছে। যাক, নিচে ফাংশনের কাজ (চিত্র-১৪) চিত্রিত হয়েছে।

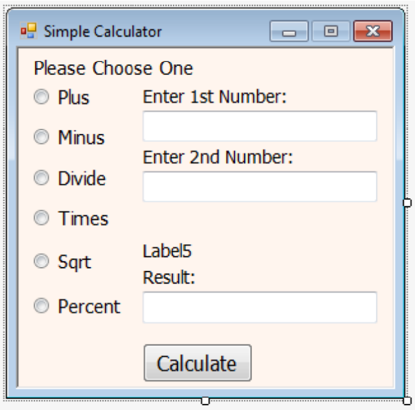
চিত্র-১৪: NumOfChar (Supply) ফাংশন -এর ব্যবহার ও ফলাফল।



ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

৩য় উদাহরণ: Sub & Function

এই উদাহরণে আমরা সত্যিকার অর্থে ব্যবহারযোগ্য একটি ক্যালকুলেটর তৈরি করতে যাচ্ছি। যে কোনো পিসিতে এটি ব্যবহার করা যাবে। প্রথমে আমরা ফর্মের ডিজাইন নিয়ে কিছু আলোচনা করবো। নিচের চিত্রে (চিত্র -১৫) এটি দেখানো হয়েছে। নিম্নে বর্ণিত পদ্ধতিতে প্রজেক্ট তৈরি করে ফর্ম ডিজাইন করুন।



চিত্র -১৫: সিম্পুল ক্যালকুলেটর প্রজেক্টের ফর্ম

1. একটি নতুন প্রজেক্ট তৈরি করুন। এর নামকরণ করুন SimpleCal
2. ফর্মের সাইজ প্রোপার্টি হবে: ৩১৫ x ৩১০। Name প্রোপার্টি হবে Simple Calculator। ফর্মের FormBorderStyle প্রোপার্টি হবে Fixed3D এবং MaximizeBox প্রেপার্টি হবে False।
3. ফর্মে ৫টি লেবেল থাকবে: label1 এর Text প্রোপার্টি হবে Please Choose One। label2 এর Text প্রোপার্টি হবে Enter 1st Number। label3 এর Text প্রোপার্টি হবে Enter 2nd Number। label4 এর Text প্রোপার্টি হবে Result:। label5 এর Text প্রোপার্টি হবে Label5- এটা দেওয়াই থাকবে।
4. ফর্মে থাকবে ৩টি টেক্সট বক্স: TextBox1, TextBox2 ও TextBox3। সবগুলোর Font Size প্রোপার্টি হবে 11। শুধু TextBox3 -এর ReadOnly প্রোপার্টি হবে True। আমরা চাইনা এই বাক্সে ব্যবহারকারী কিছু লিখবেন।
5. ফর্মে মোট ৬টি রেডিও বাটন থাকবে: এগুলোর ক্যাপশন নাম (Text property) হবে যথাক্রমে, Plus, Minus, Divide, Times, Sqrt এবং Percent। এদের (Name) প্রোপার্টি হবে যথাক্রমে, RDplus, RDminus, RDdivide, RDtimes, RDsqrt এবং RDpercent। এই প্রোপার্টি সঠিকভাবে থাকা চাই। এ নামগুলো কোডে ব্যবহৃত হবে।
6. সবশেষে ফর্মে একটি বোতাম থাকবে। এর Font Size প্রোপার্টি হবে 12 এবং Text প্রোপার্টি হবে Calculate। চাইলে ফর্মের BackColor প্রোপার্টি বদলাতে পারেন। আমি হালকা হলুদ (SeaShell) দিয়েছি।

ডিজাইন শেষ হওয়ার পর একবার Start মেনুতে ক্লিক করে রান করে দেখুন। লক্ষ করুন, ফর্ম বড় কিংবা ছোট করা যাবে না। আমরা এটাই চাই। ক্লোজ বাক্সে ক্লিক করে বদ্ধ করুন।

কোডিং

এ উদহরণে মূল কোড লিখিত হবে Calculate বোতামের Click ইভেন্টে। তবে ফর্মের Load ইভেন্টে একটি ছোট্ট কোড লেখার প্রয়োজন। আমরা চাই রেডিও বোতামের Default একটি চয়েজ থাকবে। এটি হবে Plus নামক রেডিও বোতামটি (অর্থাৎ প্রথমটি)। সুতরাং ফর্মের লোড কোড হবে নিম্নরূপ:

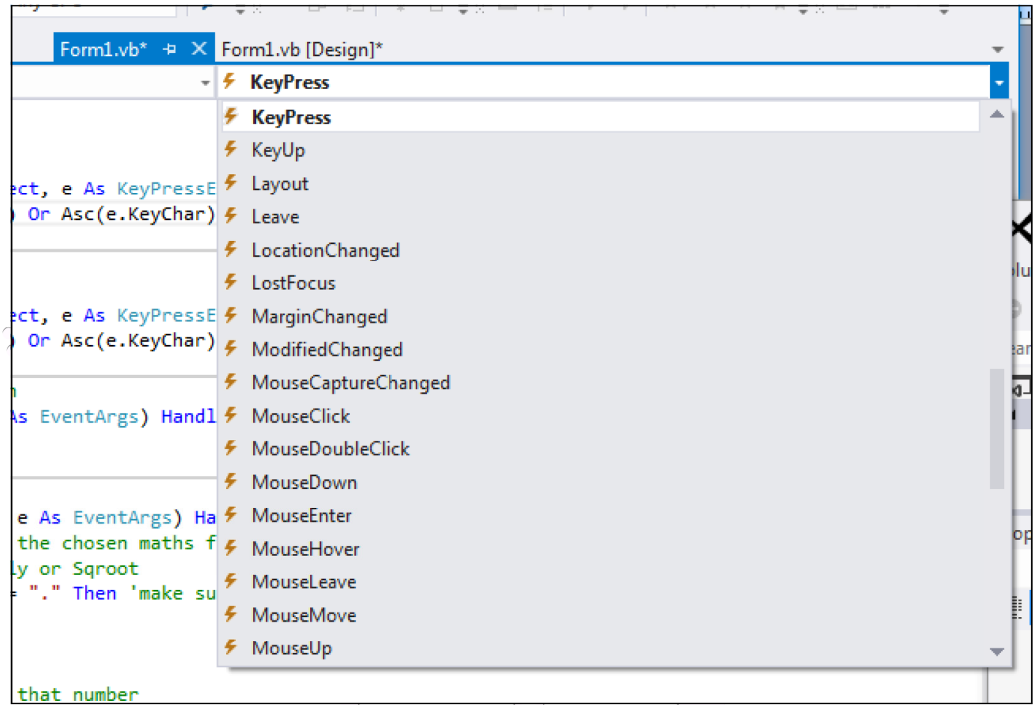
```
' default value 'Plus' -radio button1 chosen
   Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
       RDplus.Checked = True
   End Sub
```

Form1.vb [design] ক্লিক করে ফর্মের যে কোনো স্থানে [কোনো বস্তুতে নয়] ডবল ক্লিক করলেই উক্ত সাবটি আসবে। এখানে শুধু মাঝের লাইনটি লিখবেন। এই লাইন দ্বারা বুঝানো হচ্ছে: ফর্মের Plus Radio button চেক করা থাকবে। এর দ্বারা এটাও বুঝাচ্ছে যে, ব্যবহারকারী উভয় টেক্সট বাক্সে সংখ্যা লিখে Calculate বোতামে ক্লিক করলে উভয়টির যোগফল আসবে।

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

Calculate বোতামের Click ইভেন্টের কোডিংয়ে যাওয়ার পূর্বে আমরা আরো দু'টি ছোট্ট কোডিং করবো। এগুলো অনুরূপ এবং লিখিত হবে TextBox1 ও TextBox2 -এর KeyPress নামক ইভেন্টে। এই সাবটি পেতে হলে প্রথমে টেক্সটবাক্সে ডবল ক্লিক করুন। এতে টেক্সটবাক্সের TextChanged নামক ইভেন্টটি আসবে। এটা আমাদের প্রয়োজন নেই। তবে উপরে ডানদিকের মেনু থেকে চয়েজ করে নিতে হবে টেক্সটবাক্সের KeyPress নামক ইভেন্টটি (নিচরে চিত্রটি দেখুন)। নিচের সাবটি আসবে। এই সাবের মধ্যেই আমরা এক লাইন কোড লিখেছি।

চিত্র -১৬: কিভাবে টেক্সটবাক্সের KeyPress ইভেন্ট সাব বের করবেন?




```
'Make sure only numbers can be entered
Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
TextBox1.KeyPress
        e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar
= ".") And (sender.Text.IndexOf(".") = -1)))
End Sub
```

ঠিক একই কোড যাবে TextBox2 এর KeyPress ইভেন্ট সাবেও:

```
'Make sure only numbers can be entered
Private Sub TextBox2_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
TextBox2.KeyPress
        e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar
= ".") And (sender.Text.IndexOf(".") = -1)))
End Sub
```

উপরের কোড লাইনটি ভালো করে শিখে নিন। এই কোড দ্বারা যে কোনো টেক্সটবাক্সে শুধুমাত্র সংখ্যা (দশমিক একটিমাত্র বিন্দুসহ '.') এবং Delete ছাড়া অন্যকিছু যাতে কেউ লিখতে পারেন না তা নিশ্চিত করে। প্রয়োজনে কোডটি নোট বইয়ে লিখে রাখুন।

যাক এ পর্যন্ত আমরা যা করেছি তা দেখার জন্য একবার প্রোগ্রাম রান করে দেখুন। উভয় টেক্সটবাক্সে সংখ্যা লিখুন এবং সংখ্যা ছাড়া অন্য কিছু লিখার চেষ্টা করুন। এবার আমরা আসল কোড নিয়ে বিস্তারিত আলোচনা করবো।

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

Calculate বোতামের কোড

আমরা ইতোমধ্যে ফর্মের লোড ইভেন্ট ও সংখ্যা লেখার কীপ্রেস ইভেন্ট কোড নিয়ে আলোচনা করেছি। এই কোডগুলো দ্বারা প্রথমত Plus রেডিও বোতাম চয়েজ করা অবস্থা ও টেক্সটবাক্স দু'টোয় শুধুমাত্র সংখ্যা লিখা সম্ভব- তা নিশ্চিত করেছি। এবার আমরা ক্যালকুলেট বোতামের কোড নিয়ে আলোচনায় যেতে পারি। নীচে প্রথমে বোতামের কোডগুলো তুলে ধরলাম। এরপর প্রত্যেক লাইনের ওপর ব্যাখ্যা দেবো।

```
 'calculation button
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim ActWant As String = WhichCal() 'get the chosen maths fuction value
    ' ActWant= Add, Minus, Division, Multiply or Sqroot
      If TextBox1.Text = "" Or TextBox1.Text = "." Then
     Exit Sub 'make sure there is 'number in the first fld
      End If
'first fld has a number
'check if the user wants to get sqrt of that number
  If TextBox1.Text <> "" And ActWant = "Sqroot" Then
        TextBox2.Text = "" ' Note: if second textBox also has number it is
     'ignored and wiped
        Label5.Text = "√" & TextBox1.Text
        TextBox3.Text = Math.Sqrt(Val(TextBox1.Text)) 'calculate square root, display
                      'and exit
        Exit Sub
  End If
        'textBox1 has a number and chosen radio button is not sqroot
        'check if textBot2 has a valid number
      If TextBox2.Text = "" Or TextBox2.Text = "." Then 'make sure there is number in
                      'the first fld
         Exit Sub
      End If
      'check if the user trying to divide by 0
      If Val(TextBox2.Text) = 0 And ActWant = "Division" Then
          TextBox3.Text = "Division by 0 not valid!"
          Exit Sub
      End If
      'both textBox have valid numbers- calculate and display results
      If TextBox1.Text <> "" Or TextBox2.Text <> "" Then 'check again
          If ActWant = "Add" Then
              Label5.Text = TextBox1.Text & " + " & TextBox2.Text 'display info
              TextBox3.Text = Val(TextBox1.Text) + Val(TextBox2.Text) 'add and show
              Exit Sub
          End If
          If ActWant = "Minus" Then
              Label5.Text = TextBox1.Text & " - " & TextBox2.Text 'display info
              TextBox3.Text = Val(TextBox1.Text) - Val(TextBox2.Text) 'minus and show
              Exit Sub
          End If
          If ActWant = "Division" Then
             Label5.Text = TextBox1.Text & " ÷ " & TextBox2.Text 'display info
             TextBox3.Text = Val(TextBox1.Text) / Val(TextBox2.Text) 'divide and show
             Exit Sub
          End If
```

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

```
        If ActWant = "Multiply" Then
           Label5.Text = TextBox1.Text & " x " & TextBox2.Text 'display info
           TextBox3.Text = Val(TextBox1.Text) * Val(TextBox2.Text) 'multiply and show
           Exit Sub
        End If
        If ActWant = "Percent" Then
           Label5.Text = TextBox2.Text & "% of " & TextBox1.Text 'display info
           TextBox3.Text = Val(TextBox1.Text) * Val(TextBox2.Text) / 100 'get percent
                             'and show
           Exit Sub
        End If
     End If
End Sub
```

==================================== ======= ============================== =======================

উপরের কোডের ব্যাখ্য

প্রথম লাইন: `Dim ActWant As String = WhichCal()` দ্বারা আমরা `ActWant` নামক একটি ভেরিয়েবল্ ঘোষণা করছি। এটার মধ্যে একটি শব্দ থাকবে যা নির্ভর করবে ব্যবহারকারী কোন্ রেডিও বোতাম চয়েজ করেছেন তার ওপর। লক্ষ করুন একই লাইনে ভেরিয়েবলের ভ্যালু নির্ধারণ করা হয়েছে `WhichCal()` নামক ফাংশন কল করে। এই ফাংশন আমরা আলাদাভাবে ডিফাইন করেছি। এর ওপর আলোচনা একটু পরে আসছে। এখানে শুধু এটুকু জানতে হবে যে, এটা রিটার্ন করে ব্যাবহারকারী কোন্ রেডিও বোতাম চয়েজ করেছেন তার একটি নাম। সুতরাং প্রথম লাইন থেকে আমরা জানলাম ব্যবহারকারী কোন্ ক্যালকুলেশন করবেন। এবার দেখে নেই `WhichCal()` ফাংশনের কোডটি।

```
Function WhichCal() 'Returns values according to which radio button clicked
        If RDplus.Checked = True Then
            Return "Add"
        End If
        If RDminus.Checked = True Then
            Return "Minus"
        End If
        If RDdivide.Checked = True Then
            Return "Division"
        End If
        If RDtimes.Checked = True Then
            Return "Multiply"
        End If
        If RDsqrt.Checked = True Then
            Return "Sqroot"
        End If
        If RDpercent.Checked = True Then
            Return "Percent"
        End If
        Return "Add"
 End Function
```

লক্ষ করুন: এই ফাংশনের কোনো প্যারামিটার নেই। এটার কাজ হলো ব্যবহারকারী কোন্ রেডিও বোতাম ক্লিক করে রেখেছেন তা পরীক্ষা করে একটি নাম [অর্থাৎ, ‘Add’, ‘Minus’, ‘Division’, ‘Multiply’, ‘Sqroot’ কিংবা ‘Percent’] ফেরৎ পাঠানো। ফাংশনের `Return` কিওয়ার্ড এ কাজটি করে। এই কিওয়ার্ড পরে ফাংশনের কাজ শেষ হয়ে যায়। পরের লাইনে আর যাবে না।

মনে করুন, আপনি চয়েজ করলেন Times নামক রেডিও বোতামটি- অর্থাৎ পূরণ অঙ্ক করবেন। এই ফাংশন একে একে চেক করে দেখবে কোন্ রেডিও বোতাম চেক করা অবস্থায় আছে। যেতেহু `RDtimes` নামক বোতামটি চয়েজ করা আছে- তাই এখানে এসে ফাংশন রিটার্ন করবে `Multiply` ভ্যালু। অন্যকথায়, এক্ষেত্রে `ActWant` এর মান হবে `Multiply`।

৩য় পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার মৌলিক কাঠামো*

ভিজুয়্যাল বেসিক.নেট : Subs & Functions Examples

ব্যবহারকারী কোন্ হিসাব করবেন তা প্রথমেই আমরা জেনে নিলাম কোডের প্রথম লাইন থেকে। এবার একটি চেক করা চাই। দেখতে হবে প্রথম বাক্সে সংখ্যা আছে কি না। এছাড়া শুধুমাত্র দশমিক পয়েন্ট '.' তো আর সংখ্যা নয়। নিচের `If - End If` ব্লকের কাজ এটি। সংখ্যা না থাকলে কিংবা শুধু '.' থাকলে সাব থেকে বেরিয়ে যাবে।

```
If TextBox1.Text = "" Or TextBox1.Text = "." Then 'make sure there is number in the first fld
   Exit Sub
End If
```

যদি গ্রহণযোগ্য সংখ্যা প্রথম টেক্সট্বাক্সে থাকে তাহলে পরবর্তী ব্লকের কাজ শুরু হবে। এটা প্রথমে দেখে নেয় কোন্ অঙ্ক করা হবে- তারপর সিদ্ধান্ত নেয়। প্রথম টেক্সট্বাক্সে সংখ্যা থাকলে ও ব্যবহারকারী এটির স্কোয়ার রুট বের করতে ইচ্ছুক হলে ব্লকের ভেতরের লাইনগুলো কাজ করে সাব থেকে বেরিয় যাবে। মনে রাখবেন- স্কোয়ার রুটের জন্য সংখ্যা লাগে মাত্র একটি। সুতরাং দ্বিতীয় বাক্সে কিছু আছে কি না তা চেক করা নিষ্প্রোজন।

```
If TextBox1.Text <> "" And ActWant = "Sqroot" Then
     TextBox2.Text = "" ' Note: if second textBox also has number it is ignored and wiped
     Label5.Text = "√" & TextBox1.Text
     TextBox3.Text = Math.Sqrt(Val(TextBox1.Text)) 'calculate square root display and exit
     Exit Sub
 End If
```

যদি প্রথম টেক্সট্বাক্সে সংখ্যা থাকে এবং ব্যবহারকারী চান অন্য কোনো হিসাব করতে তাহলে দেখতে হবে দ্বিতীয় টেক্সট্বাক্সে আরেকটি সংখ্যা আছে কি না। এই কাজটি করে পরের কোড ব্লকটি।

```
If TextBox2.Text = "" Or TextBox2.Text = "." Then 'make sure there is number in the second fld
    Exit Sub
End If
```

দ্বিতীয় টেক্সট্ বাক্সে গ্রহণযোগ্য সংখ্যা আছে তা নিশ্চিত হলেই পরবর্তী কোড ব্লক কাজ শুরু করবে। এটার কাজ হলো একটি বিশেষ কন্ডিশন চেক করা। আমরা জানি যে কোনো সংখ্যাকে ০ দ্বারা ভাগ করা যায় না। ব্যবহারকারী ঠিক এটি করছেন কি না তা চেক করে দেখে নিচের এই কোড ব্লকটি।

```
If Val(TextBox2.Text) = 0 And ActWant = "Division" Then
   TextBox3.Text = "Division by 0 not valid!"  'এই লাইনটি ডিসপ্লে ফিল্ডে প্রদর্শিত হবে যাতে ব্যবহারকারী বুঝতে পারেন।
   Exit Sub
End If
```

এ পর্যন্ত সকল চেক-টেক সঠিক মতো হয়ে গেলে আমরা নিশ্চিত হবো ব্যবহারকারী +, -, X, ÷ কিংবা % - এই ৫টির মধ্যে যে কোনো একটি হিসাব করবেন। আর কোন্টি করবেন তা সংরক্ষিত আছে ঐ প্রথম লাইন থেকে পাওয়া ActWant ভেরিয়েবলে। সুতরাং বাকি ৫টি ব্লকের কাজ হলে ActWant ভেরিয়েবলে সংরক্ষিত তথ্য মুতাবিক হিসাব করে প্রদর্শন করা। অন্য কথায় ভেরিয়েবলে যে নামটি আছে সে মুতাবিক ব্লক চয়েজ করে হিসাব করবে। আমরা এই সবগুলো ব্লক আর এখানে দেখাচ্ছি না- শুধু একটা দেখলেই বুঝতে সক্ষম হবো কি হচ্ছে। মনে করুন ভাগ হবে- তাহলে নিচের কোড ব্লকটি সক্রিয়া হয়ে প্রোগ্রাম শেষ করবে।

```
 If ActWant = "Division" Then
     Label5.Text = TextBox1.Text & " ÷ " & TextBox2.Text -এই লাইন দ্বারা তথ্য দেওয়া হচ্ছে কোন্ হিসাব হবে।
     TextBox3.Text = Val(TextBox1.Text) / Val(TextBox2.Text) 'divide and show
     Exit Sub
 End If
```

ব্লকের দ্বিতীয় লাইন দ্বারা আমার বুঝিয়েছি ব্যবহারকারী উভয় সংখ্যা দিয়ে কোন্ হিসাব করবেন। এখানে বলা হচ্ছে: লেবেল৫-এর টেক্সট্ হবে প্রথম সংখ্যা ভাগ দ্বিতীয় সংখ্যা। অনুরূপ অন্যান্য ক্ষেত্রেও এই তথ্য দেওয়া থাকবে। ব্যস! প্রোগ্রামের কোড নিয়ে আলোচনার ইতি হলো। ব্যাখ্যাগুলো প্রয়োজনে পুনরায় পাঠ করুন। কোডের একশন বুঝতে পারলে আপনি অনেকদূর এগিয়ে গেলেন।

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

আমরা ইতোমধ্যে ভিজুয়্যাল বেসিক.নেট এর কয়েকটি উদাহরণ দেখেছি। এই পরিচ্ছেদেও আরো কিছু উদাহরণ দেখবো। তবে এসব উদাহরণের উদ্দেশ্য হবে ভাষার Variable Declaration এবং Data type সম্পর্কে অবগত হওয়া।

Variable কি?

যে কোনো প্রোগ্রামিং ভাষায় পরিবর্তনযোগ্য কিছু তথ্যের প্রয়োজন পড়ে। যে উপায়ে এটা সম্ভব তাহলো কম্পিউটারের মেমোরিতে তথ্যের ধরন চিহ্নিত করে বিশেষ নাম দ্বারা জায়গা সংরক্ষণ করা। আর প্রয়োজনে প্রোগ্রাম থেকে ঐ তথ্যটুকু এনে এর উপর কাজ, রদবদল, হিসাব-নিকাশ, পরিবর্তন ইত্যাদি করে পুনরায় সংরক্ষণ করা। এছাড়া দৃশ্যমান বস্তু যেমন টেক্সট বাক্স, লেবেল, লিস্ট বাক্স ইত্যাদিতে প্রদর্শন করা। মোটকথা ভেরিয়েবল ছাড়া কোনো কম্পিউটার প্রোগ্রাম লিখা প্রায় অসম্ভব। সুতরাং এটা কিভাবে ডিক্লায়ার করতে হয় এবং কোন্ উপায়ে কাজে লাগানো যায় তা জানা আমাদের জন্য একান্ত জরুরী।

কিভাবে ভেরিয়েবল ডিক্লারেশন করবেন?

প্রত্যেক ভেরিয়েবল নম্নিলিখিত উপায়ে ডিক্লায়ার করতে হবে:

১. স্কোপ বা ব্যবহারের ব্যপ্তি: ফর্মের কোথায় এটা ডিক্লায়ার করছেন তা-ই নির্ধারণ করে তার স্কোপ। ফর্মের ক্লাস ডিক্লারেশনের পরই (অর্থাৎ প্রথম লাইন বাদে) ডিক্লায়ার করলে ক্লাসের ভেতরস্থ সকল প্রসিডোর, সাব, ফাংশন ইত্যাদিতে এটা কাজে লাগানো যায়। অপরদিকে যে কোনো প্রসিডোর, সাব ও ফাংশনের ভেতর ডিক্লায়ার করলে শুধু ঐ সাবের ভেতরই এটা ব্যবহারযোগ্য হবে। নিচের উদাহরণ দু'টো দেখুন।

উদাহরণ ১. সর্বত্র ব্যবহারযোগ্য ভেরিয়েবল।

```
Public Class Form1
   Dim intNum As Integer 'intNum scope throughout rest of the program
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      intNum = 20 'Here intNum = 20
   End Sub
   Private Sub TextBox1_MouseLeave(sender As Object, e As EventArgs) Handles TextBox1.MouseLeave
      intNum = 50 'Here intNum = 50
   End Sub
   Function NumOfChar()
      intNum = Len(TextBox1.Text) 'Here intNum = Number of characters in TextBox1
      Return intNum
   End Function
End Class
```

উদাহরণ ২. শুধুমাত্র সাব বা ফাংশনের ভেতর ব্যবহারযোগ্য।

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        dim EntText as String = TextBox1.Text 'only available in this sub
    End Sub
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        MsgBox(EntText) 'will not work -EntText is not declared error comes- because here it isn't useful.
    End Sub
End Class
```

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

Data Type

কিম্পউটারে বুঝার জন্য যে কোনো তথ্যের ধরন বা টাইপ কি তা বলে দিতে হবে। ভেরিয়েবল ঘোষণার সময় এর Type-ও লিখে দিতে হয়। যেমন:

```
Dim MyVariable as String 'String is the type
Dim MyVariable as Integer 'Integer is the type
Dim MyVariable as Single 'Single (decimal number) is the type
```

এই গ্রন্থে ব্যবহৃত বিভিন্ন Type

1. Boolean - True or False
2. Char (single Character) - 0 through 65535 (unsinged)
3. Date - 0:00:00 (midnight) on January 1, 0001 through 11:59:59 PM on December 31, 9999
4. Double (double precision floating point number)
5. Single (single precision floating point number)
6. Integer - -2,147,483,648 through 2,147,483,647 (signed)
7. Long (long Integer)
8. Object - Any type can be stored in a variable of type Object
9. Short (short integer) - -32,768 through 32,767 (signed)
10. String - 0 to approximately 2 billion Unicode characters

এবার আমরা উপরোক্ত প্রতিটি টাইপ ডিক্লারেশন ও ব্যবহার পদ্ধতির উদাহরণ দেখবো।

1. Boolean

```
Dim RadButOn As Boolean
' Check to see if RadioButton1 is ON or OFF.
If RadioButton1.Checked = True Then
        RadButOn = True
    Else
        RadButOn = False
End If
```

2. Char

```
'Normally we don't need this type. It is used to hold a single character only.
Dim MyChar as Char
MyChar = "A"
```

3. Date

```
 Dim date1 As Date = Date.Now
'gives local date and time: ex. 6/21/2016 1:34:14
 Dim date2 As Date = Date.UtcNow
'gives short date and coordinated universal time (GMT time without daylight change: ex. 6/21/2016 7:34:14)
 Dim date3 As Date = Date.Today
'gives just the short date: ex. 6/21/2016
```

ভিজুয়াল বেসিক.নেট : *Variable & Data Type*

4. Double

```
Dim DecOne as Double = 23.09898731
Dim DecTwo as Double = 5.98987654
MsgBox(DecOne * DecTwo) 'gives 138.360082185927
```

5. Single

```
Dim DecOne as Single = 23.09898731
Dim DecTwo as Single = 5.98987654
MsgBox(DecOne * DecTwo) 'gives 138.3601
```

উপরোক্ত Double এবং Single টাইপের মধ্যে পার্থক্য লক্ষ করুন। প্রথমটি দ্বিতীয়টি থেকে অধিক সঠিক।

6. Integer

```
Dim MyNum as Integer
```

লক্ষ করুন: ছোট আস্ত সংখ্যা নিয়ে কাজ করতে ইন্টেজার টাইপ ব্যবহার করা উত্তম।

7. Long

```
Dim MyNum as Long
```

লক্ষ করুন: বড় আস্ত সংখ্যা নিয়ে কাজ করতে লং টাইপ ব্যবহার করা উত্তম।

8. Object

```
'Any type can be stored in an Object Data Type
Dim v As Object
v = "17"
' v contains the 2-character String value "17".
v = v - 15
' v now contains the Integer value 2.
v = "H" & v
' v now contains the 2-character String value "H2"
```

9. Short

```
Dim ANum as Short 'Use short data type if you work with small integer to save memory
```

10. String

```
Dim j As String = "Joe said ""Hello"" to me."
Dim h As String = "Hello"
' The following messages all display the same thing:
' "Joe said "Hello" to me."
MsgBox(j)
MsgBox("Joe said " & """" & h & """" & " to me.")
MsgBox("Joe said """ & h & """ to me.")
```

সাধারণত আমরা Single, Long এবং Short Data type ব্যবহার করবো।

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

Array Variables

এ পর্যন্ত আমরা যেসব ভেরিয়েবল নিয়ে আলোচনা করেছি সেগুলোর প্রতিটিই মূলত একক কোনো ভ্যালু সংরক্ষণ করার ক্ষমতা রাখে। যেমন:

Dim A as integer 'A যে কোনো একটিমাত্র সংখ্যা সংরক্ষণ করতে পারে। তবে এটা পরিবর্তনশীল।

Dim Astring as string 'Asring যে কোনো একটিমাত্র string সংরক্ষণ করতে পারে। যেমন: Asring = "Ab"

মোটকথা, সাধারণ সকল ভেরিয়েবল যে কোনো একটি তথ্য সংরক্ষণ করতে পারে এবং তা পরিবর্তনশীল। তবে আপনি যদি চান ১০টি, ২০টি, ১০০টি কিংবা আরো বেশি সংখ্যা, নাম ইত্যাদি সংরক্ষণ করবেন তাহলে উপায় কি? প্রতিটি ভ্যালুর জন্য একেকটি নাম ডিক্লায়ার করতে পারেন। কিন্তু এভাবে ক'টি করবেন? শীঘ্রই আপনি প্রোগ্রাম করা থেকে ইস্তিফা দেবেন! ভাবার কোনো কারণ নেই। মাইক্রসফট ভিজুয়্যাল বেসিক আমাদেরকে শত শত বা এমনকি হাজার হাজার ভিন্ন সংখ্যা, নাম ইত্যাদি খুব সহজে ডিক্লায়ার করে সংরক্ষণের ব্যবস্থা করে দিয়েছে। এই ব্যবস্থার নামই হলো: Array Variables

Array variable declaration

এই ভেরিয়েবল ঘোষাণার নিয়ম এলো এরূপ:

```
Dim A(7) As Integer  'Here A is one-dimensional array with 8 elements.
```

লক্ষণীয় সকল অ্যারের উপদান ০ থেকে শুরু হয়। সুতরাং আপনি যদি আটটি ভিন্ন সংখ্যা বা অন্য কিছু ডিক্লায়ার করতে চান তাহলে ব্র্যাকেটের ভেতর ৭ লিখবেন। ১০টি চাইলে লিখবেন ৯, ১৫টি হলে লিখবেন ১৪। উপরের ডিক্লারেশনের ফলে যে ৮টি ইলেমেন্ট তৈরি হবে তাহলো নিম্নরূপ:

A(0), A(1), A(2), A(3), A(4), A(5), A(6), A(7)

কিভাবে ভ্যালু প্রবেশ করাবেন? কয়েকটি উপায় আছে। নিচে এগুলো উদাহরণ দ্বারা দেখিয়েছি।

১. মনে করুন আপনি ০ থেকে ৫ -এই ৬টি সংখ্যা অ্যারেতে সংরক্ষণ করবেন। আপনি এভাবে করবেন:

```
Dim MyNum (5) as Integer 'define array
Dim i as integer 'define a counter
For i = 0 to  5
   MyNum (i) = i
Next
```

এই প্রোগ্রাম অংশ রান করার পর MyNum নামক অ্যারে ভেরিয়েবলে থাকবে:

MyNum (0) = 0, MyNum (1) = 1, MyNum (2) = 2, MyNum (3) = 3, MyNum (4) =4 এবং MyNum (5) = 5

২. মনে করুন আপনি ব্যবহারকারীর কাছে ১০টি নাম লিখতে বলবেন এবং এগুলো একটি অ্যারে ভেরিয়েবলে সংরক্ষণ করবেন। আপনি তা এভাবে কতে পারেন:

```
Dim My10Names(9) As String ' define the array
Dim C As Integer ' define a counter
' fill the array
For C = 0 To 9
   My10Names(C) = InputBox("Enter the name please: " & C)
Next
```

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

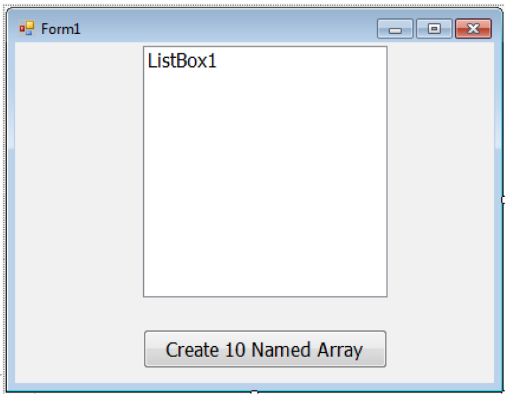
ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

সংরক্ষিত নামগুলো এক সঙ্গে দেখতে চাইলে আমরা ব্যবহার করতে পারি একটি লিস্টবক্স। নিম্নের কোড অংশ উপরের কোডের নিচে লিখলেই হবে।

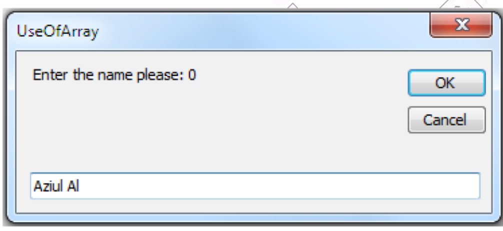
```
ListBox1.Items.Clear() ' display the result
For I = 0 To 7
    ListBox1.Items.Add(My10Names(I))
Next
```

Test 1: নতুন একটি প্রজেক্ট তৈরি করে এতে একটি লিস্টবক্স ও একটি বোতাম দিন। বোতামের নাম দিন ‘Create 10 Names Array’। বোতামের ক্লিক ইভেন্টে উপরের কোড লিখে রান করুন। লিস্ট বাক্সে সব নাম এসছে কি? সমস্যা হলে নিচের ছবিগুলো দেখুন।

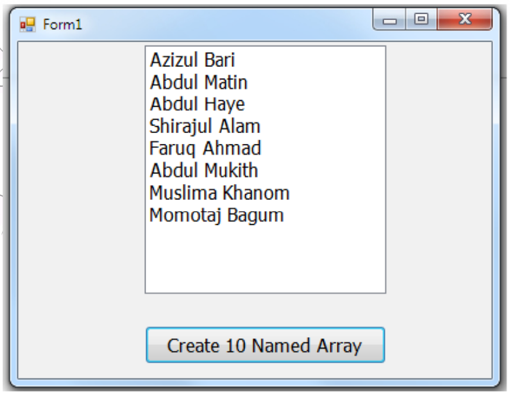
```
Public Class Form1
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
      Dim My10Names(9) As String 'define thearray
      Dim C As Integer ' define a counter
      ' fill the array
      For C = 0 To 9
         My10Names(C) = InputBox("Enter the name please: " & C)
      Next
      ListBox1.Items.Clear() ' display the result
      For I = 0 To 7
         ListBox1.Items.Add(My10Names(I))
      Next
   End Sub
End Class
```



চিত্র -১৭: ফর্ম ডিজাইন



চিত্র -১৮: নাম প্রবেশের ডায়লগ



চিত্র -১৯: রানের পর লিস্ট

মাইক্রসফট ভিজুয়াল বেসিক.কম উপরোক্ত ‘Array’ তৈরির সুযোগ দিয়ে অনেক তথ্য সংরক্ষণ ও ব্যবহারের ব্যবস্থা করে দিয়েছে। তবে এখানেই শেষ নয়। এ্যারে ছাড়াও আরো একটি গুরুত্বপূর্ণ তথ্য সংরক্ষণ ও ব্যবহার পদ্ধতি আবিষ্কার করেছে। এটা এ্যারে থেকেও উন্নতমানের এবং সহজ। এই ব্যবস্থার নাম collection। আমরা এখন এ সম্পর্কে বিস্তারিত আলোচনায় যাচ্ছি।

৪র্থ পরিচ্ছেদ: *ভিজুয়্যাল বেসিক.নেট ভাষার* ***Variable Declaration***

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

Collection Declaration & it's use -কালেকশন ঘোষণা ও এর ব্যবহার

Collection অনেকটা এ্যারের মতোই কাজ করে। তবে, কালেকশনের মধ্যে একাধিক মান বা ভেরিয়েবল্ সংরক্ষণ ও ব্যবহার করা যায় একই সঙ্গে। নিচে এ্যারে ও কালেকশের মধ্যে মূল ক'টি পার্থক্য তুলে ধরেছি।

ক. এ্যারের ইনডেক্স শুরু হয় ০ থেকে। কিন্তু কালেকশনের ইনডেক্স শুরু হয় ১ থেকে। যেমন:

Dim AnArray(3) - তে চারটি ইলেমেন্ট বা তথ্য থাকবে: AnArray(0), AnArray(1), AnArray(2) ও AnArray(3)। অপরদিকে:

```
Dim MyCollection as New Collection
MyCollection.add (1, "Sharif")
MyCollection.add (2, "Nasim")
MyCollection.add (3, "Abdullah")
MyCollection.add (4, "Khaled")
```

উপরের কালেকশনে চারটি ইলেমেন্ট সংরক্ষিত হবে। আমরা এদেরকে ব্যবহার করবো এভাবে রেফারেন্স করে: MyCollection(1), MyCollection(2), MyCollection(3) ও MyCollection(4)।

খ. আরেক পার্থক্য হলো ভেরিয়েবল টাইপ। এ্যারের মধ্যে সকল ইলেমেন্ট হতে হবে একই ধরনের। যেমন:

Dim MyArray (0 to 5) as integer -এখানে ০ থেকে ৫ পর্যন্ত ৬টি আস্ত সংখ্যা ঘোষণা করা হয়েছে।

Dim MyArray ("Aziz", "Matin", "Zakir") as string -এখানে তিনটি নাম যাদের সবার ভেরিয়েবল টাইপ হলো স্ট্রিং ঘোষণা করা হয়েছে। অপরদিকে কালেকশনের ইলেমেন্টগুলো বিভিন্ন ধরনের হতে পারে। যেমন:

```
Dim MyCollection as New Collection
MyCollection.add ("Sharif")
MyCollection.add (2)
MyCollection.add ("Hello!")
MyCollection.add ("01732562315")
```

গ. আরেক পার্থক্য হলো কালেকশনের ইলেমেন্ট সরাসরি যুক্ত ও সরানো যায়। অপরদিকে এ্যারে ফিক্স। যেমনে তেমনে ভ্যালু পরিবর্তন করা যায় না।

ঘ. আরো একটি পার্থক্য হলো, এ্যারে ইলেমেন্ট সহজেই পরিবর্তন বা অভাররাইট করা যায়। যেমন:

```
Dim MyArray (2)
MyArray (0) = 5
MyArray (0) = 7
```

কালেকশনের সময় এটা করা যাবে না। তবে অন্যভাবে করার ব্যবস্থা আছে। পরে আমরা সে সম্পর্কে আলোচনা করবো।

কালেকশনের কয়েকটি মেথড আছে যেমন:

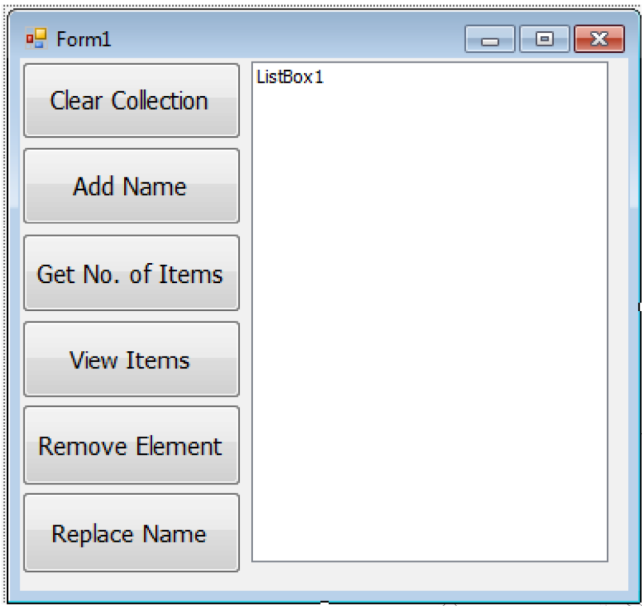
```
Dim C As New Collection 'Declaration
C.Add (1, "Aziz") 'adding element
C.Add (2, "Hello!")
C.Remove(1) 'remove element in index 1
I = C.Count 'get number of elements in C as integer
```

এখন আমরা একটি সাধারণ প্রোগ্রাম করবো। এতে কালেকশনের ব্যবহার সুস্পষ্ট হবে।

ভিজুয়্যাল বেসিক.নেট : ***Variable & Data Type***

CollectionDemo Application

১. প্রথমে CollectionDemo নামকরণে একটি ফর্ম প্রজেক্ট তৈরি করে নিচের চিত্রের মতো ডিজাইন করুন। এতে থাকবে ৬টি বোতাম ও ১টি লিস্ট বাক্স। দেখুন নিচের চিত্রটি (চিত্র: ২০)।



চিত্র -২০: CollectionDemo প্রজেক্টের ফর্ম

লক্ষ করুন, প্রতি বোতামের নামকরণ হবে দেখানো নামগুলোর মতো। লিস্ট বাক্সের নাম যা আছে তাই থাকবে। এবার নিচের কোডগুলো বর্ণিত বোতামের ক্লিক ইভেন্টে লিখুন।

১. Collection Declaration: এটা হবে ফর্মের ক্লাস ডিক্লারেশনের পরে:

```
Public Class Form1
    Dim MyCollection As New Collection
End Class
```

২. যথাক্রমে ‘Clear Collection’, থেকে ‘Replace Name’ বোতামের ক্লিক ইভেন্ট কোডসমূহ:

```
Public Class Form1
    Dim MyCollection As New Collection
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        ' this method clears all the elements in the collection
        MyCollection.Clear()
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        ' read a name
        Dim Name As String
        Name = InputBox("Enter a Name") 'ask user to enter a name
        ' add the name into the list
        MyCollection.Add(Name)
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
        'Show number of items in a message box.
        MsgBox("The number of items is: " & MyCollection.Count)
    End Sub
```

ভিজুয়্যাল বেসিক.নেট : *Variable & Data Type*

CollectionDemo Application

```
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
        ' clear old content
        ListBox1.Items.Clear()
        ' insert the items into the list box
        Dim I As Integer
        For I = 1 To MyCollection.Count
           ListBox1.Items.Add(MyCollection(I))
        Next
End Sub

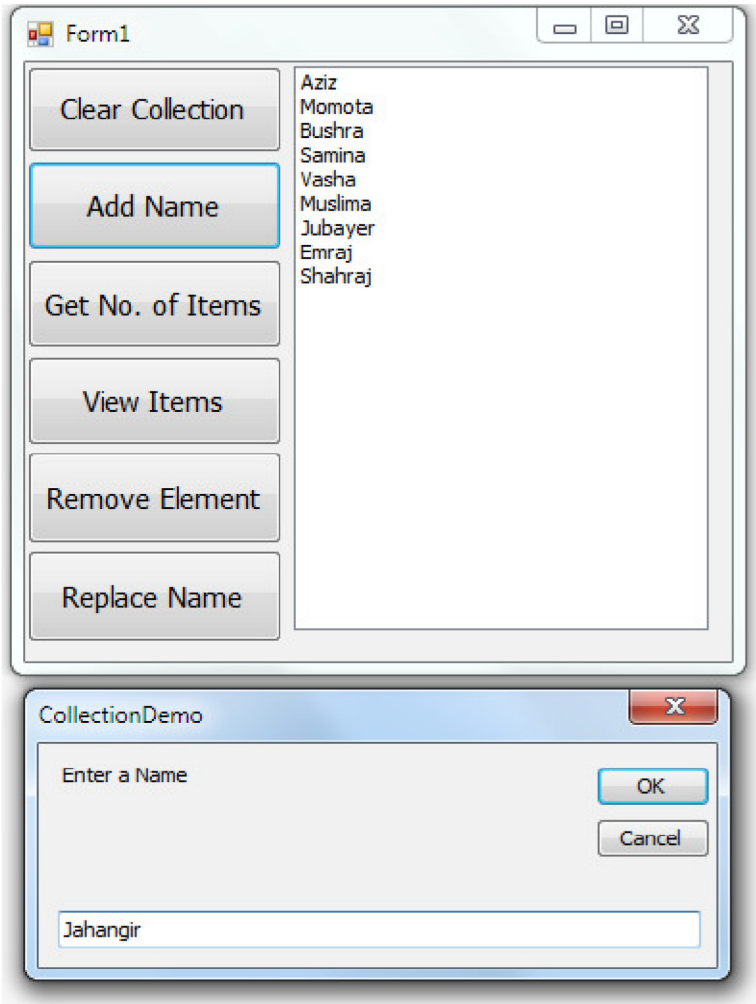
Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click
        ' get element position
        Dim I As Integer
        I = InputBox("Enter the element number you want to remove:") 'ask user to enter
element number
        If I <= MyCollection.Count And I >= 1 Then 'check in the element number is valid
           MyCollection.Remove(I)
        Else
           'if invalid number enrered then show a message, reminding of number of elements
           MsgBox("You have 1 to " & MyCollection.Count & " items in the collection.")
        End If
 End Sub

 Private Sub Button6_Click(sender As Object, e As EventArgs) Handles Button6.Click
        ' get element position
        Dim I As Integer
        Dim N As String
        I = InputBox("Enter the element number:") 'ask user to enter element number
        If I <= MyCollection.Count And I >= 1 Then 'check in the element number is valid
           N = InputBox("Enter the new name:") 'ask user to enter new name
           MyCollection.Add(N, , , I) 'add name in the element number
           MyCollection.Remove(I) 'remove old element
        Else
           'if invalid number enrered then show a message, reminding of number of elements
           MsgBox("You have 1 to " & MyCollection.Count & " items in the collection.")
        End If
    End Sub
End Class
```

উপরের কোড দ্বারা আমরা কালেকশন ডিক্লায়ার করা, ক্লিয়ার করা, নতুন ইলেমেন্ট যোগ দেওয়া, ক'টি ইলেমেন্ট আছে তা জানা, সবগুলো প্রদর্শন করা, মুছে দেওয়া এবং পরিবর্তন করার পদ্ধতি দেখিয়েছি। পরের পৃষ্ঠায় প্রোগ্রাম রানিং অবস্থার একটি চিত্র (চিত্র : ২১) দিয়েছি। একটু দেখে নিন।

ভিজুয়্যাল বেসিক.নেট : ***Variable & Data Type***

CollectionDemo Application



চিত্র -২১: CollectionDemo Application চলাকালীন দৃশ্য।

আমি আশাবাদী আলোচিত প্রোগ্রামটি কিভাবে কাজ করে তা আপনি সম্পূর্ণরূপে বুঝতে সক্ষম হয়েছেন। তবে, কোনো সমস্যা হলো পুরো পরিচ্ছেদটি পুনরায় পাঠ করুন। সকল কোডে ইংরেজিতে কমেন্ট দেওয়া আছে। এগুলো পাঠ করলে বুঝতে অনেকটা সহজ হবে।

এখন আমরা প্রাথমিক চারটে পরিচ্ছেদে কি শিখলাম তার সারসংক্ষেপ তুলে ধরছি।

১. পরিচ্ছেদ ১ তথা ভূমিকা থেকে আমরা জেনেছি ভিজুয়্যাল বেসিক.নেট ভাষার মাধ্যমে কি কি করা যায় এবং এ ভাষার মৌলিক গুণাগুণ ইত্যাদি।

২. দ্বিতীয় পরিচ্ছেদে আমরা শিখেছি:

ক. আই.ডি.ই সম্পর্কে।

খ. বিসমিল্লাহ প্রোগ্রাম।

৩. তৃতীয় পরিচ্ছেদে আমরা শিখেছি ভিজুয়্যাল বেসিক.নেট এর প্রোগ্রামিং কাঠামোর মৌলিক দু'টি বিষয়: Subroutine (Sub) এবং Function কিভাবে কাজ করে। আমরা উভয়টির কোডিং নিয়ম বা Syntax দেখেছি। প্রাকটিস প্রোগ্রাম নিয়ে আলোচনাও হয়েছে। আমরা Simple Calculator Application বানিয়েছি। এটা ছিলো একটি ব্যবহারযোগ্য প্রোগ্রাম। এ প্রোগ্রাম দ্বারা ৬টি পাটিগণিতের হিসাব করা যায়। এই প্রজেক্টের ডিজাইন ও কোডের ওপর বিস্তারিত ব্যাখ্যাও দেওয়া হয়েছে। আমি আশা করি আপনি এই প্রজেক্ট পাঠ করে কোডিং সম্পর্কে বেশ অভিজ্ঞ হয়ে ওঠেছেন।

৪. সবশেষে বর্তমান এই পরিচ্ছেদে আমরা ভিজুয়্যাল বেসিক.কম -এ ব্যবহারযোগ্য ভেরিয়েবল নিয়ে বিস্তারিত আলোচনা হয়েছে। আমরা দু'টি গুরুত্বপূর্ণ ভেরিয়েবল সম্পর্কেও উদাহরণসহ তথ্যাদি তুলে ধরেছি: ১. Array এবং ২. Collection

পরের পরিচ্ছেদে Looping নিয়ে আরো কিছু আলোচনা সংক্ষিপ্তভাবে করবো। এরপর আমরা ভিজুয়্যাল বেসিক.নেট প্রোগ্রামে সচরাচর ব্যবহৃত একটি ব্যবহারকারী দ্বারা তৈরি-করা ভেরিয়েবল নিয়ে আলোচনা করবো উদাহরণসহ। এই ভেরিয়েবলের নাম Structure।

৫ম পরিচ্ছেদ: *Looping in Program*

বিভিন্ন লুপিং : *Various Looping*

আমরা ইতোমধ্যে ভিজুয়্যাল বেসিক.নেট এর কোডিংয়ে For .... Next লুপিং এর ব্যবহার দেখেছি। এটার আরেকটি উদাহরণ হলো:

```
Dim I As Integer
For I = 1 To 20
    ListBox1.Items.Add(I)
Next
```

এই ফর-নেক্সট লুপ ১ থেকে ২০ পর্যন্ত সংখ্যা লিস্ট বাক্স ১-এ প্রদর্শন করবে। প্রত্যেকবার লুপিংয়ের সময় I-এর মাত্রা ১ বাড়তে থাকে। মাঝের লাইন এই আই I-এর মাত্রা লিস্ট বাক্সে লিখে দেয়। I-এর মান ২০ হওয়ার পর আরেকবার লুপিংয়ের জন্য ফিরে আসলে I-এর মান হবে ২১। সুতরাং লুপিং শেষ হবে, কারণ নির্দেশ দেওয়া আছে I হবে ১ থেকে ২০ -এর বেশিও না কমও না। এবার আমরা আরেক ধরনের লুপিং নিয়ে আলোচনা করবো।

Do Loop

এই লুপের নিয়ম নিম্নরূপ:

```
Do
   statement .....
   statement....
Loop While condition
```

অথবা

```
Do
   statement.....
   statement....
Loop Until condition
```

অথবা

```
Do While condition
   statement...
   statement...
Loop
```

অথবা

```
Do Until Condition
   statement...
   statement...
Loop
```

লক্ষ করুন: কন্ডিশন কোথায় আছে তা-ই নির্ধারণ করে কখন চেক করা বা বের হওয়া কিংবা থাকা নির্দিষ্ট বৈধ হবে। যেমন:

৫ম পরিচ্ছেদ: *Looping in Program*

বিভিন্ন লুপিং : *Various Looping*

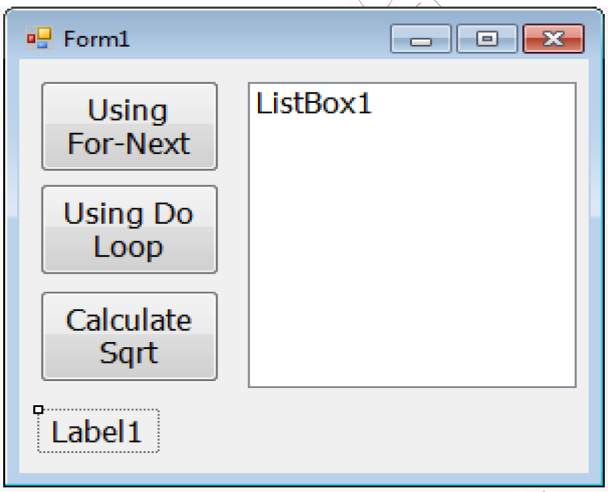
```
Do While L<10
    statement ...
    statement ...
Loop
```

এখানে প্রত্যেকবার চেক হবে লুপের কাজ শুরুর পূর্বে। এই লুপ কখনো শুরু না হওয়ার সম্ভাবনা আছে।

```
Do
    statement ...
    statement ...
Loop While L<10
```

এখানে চেক হবে লুপের কাজ শুরুর পরে। এই লুপ অন্তত একবার শুরু হবে।

এবার একটি উদাহরণ দেখে নিই। নিচের ফর্মের মতো একটি ফর্ম ডিজাইন করুন। এতে থাকবে তিনটি বোতাম, একটি লিস্ট বাক্স ও একটি লেবেল।



চিত্র -২২: লুপ উদাহরণ ফর্ম ডিজাইন

প্রথম বোতামের কোড হবে:

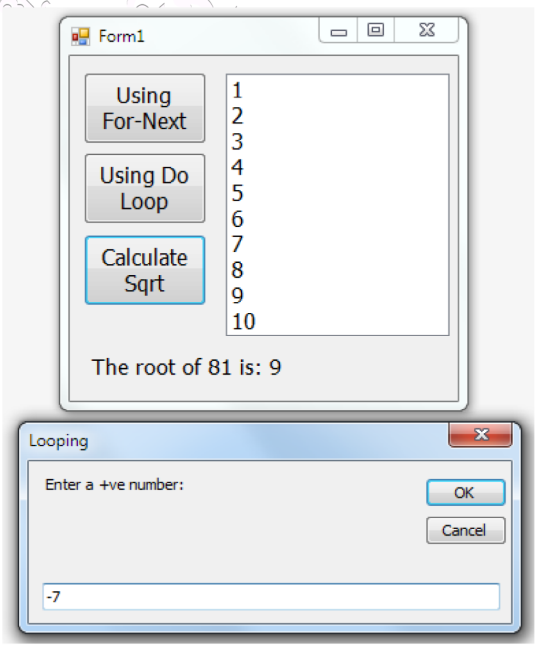
```
Dim I As Integer 'declare an integer variable
 ListBox1.Items.Clear() 'empty out list box
 For I = 1 To 10
    ListBox1.Items.Add(I) '1 to 10 into list box
Next
```

দ্বিতীয় বোতামের কোড হবে:

```
 Dim I As Integer 'declare integer variable
 I = 1 'initialize it
 ListBox1.Items.Clear() 'clear items in the list box
  Do
     ListBox1.Items.Add(I) 'add 1 to 10 in the list box
         I = I + 1 'add 1 to I
  Loop While (I < 11) 'check. I must be 11 to exit the loop
```

তৃতীয় বোতামের কোড হবে:

```
Dim I As Double 'declare double type variable (decimal number).
Do
I = InputBox("Enter a +ve number:") 'ask for a positive number.
'calculate sqrt and display in the label
    If I >= 0 Then Label1.Text = "The root of " & I  & " is: " &
Math.Sqrt(I)
Loop
```



চিত্র -২৩: লুপ উদাহরণ রানটাইম

লক্ষ করুন: প্রথম দু’টি বোতামের লুপ কোড এমনিতেই শেষ হবে। কিন্তু তৃতীয় বোতামের লুপ কোড ব্যবহারকারী নেগেটিভ সংখ্যা দ্বারা শেষ করবেন। অন্যথায় এই লুপ চিরকাল চলতে থাকবে! একে বলে নেভার এন্ডিং লুপ [Never ending loop]। প্রোগ্রামারকে এ ব্যাপারে সতর্ক থাকা চাই। অনাকাঙ্ক্ষিত নেভার এন্ডিং লুপে চলে গেলে কম্পিউটার ক্রাশ করতে পারে। যেমন আমরা যদি লিখে ফেলি (এটা করবেন না!):

```
dim a as integer =1
For b = 1 to a
    a = a+1
Next b
```

এখানে মাঝের লাইন দ্বারা a-এর মাত্রা এক এক করে বাড়তে থাকবে। সুতরাং b-এর মাত্রা কখনও শেষ হবে না এবং লুপও ক্লোজ হবে না। কম্পিউটার ক্রাশ করবে। উপরের লুপে `If I >= 0 Then` না থাকলে প্রোগ্রাম বন্ধ করা যেতো না।

৬ষ্ট পরিচ্ছেদ: *Structure*

বিভিন্ন ভেরিয়েবল্ একত্রে : *Different Variables together*

একাধিক ভিন্ন ভেরিয়েবল্ নিয়ে একেকটি Structure তৈরি হয়। এটা মূলত প্রোগ্রামার দ্বারা বিভিন্ন ভেরিয়েবল একটি ইউনিট হিসাবে সৃষ্টি করার সুযোগ যা অনেকটা সুবিধাসম্পন্ন। উদাহরণ দ্বারা বিষয়টি পরিষ্কার হবে।

মনে করুন আপনি কোনো কোম্পানির কর্মচারীদের বিভিন্ন তথ্য একত্রে সংরক্ষণ করতে চান। তথ্যাদি হবে নাম, ঠিকানা, টেলিফোন নাম্বার, বয়স, বেতন, কাজের নাম ইত্যাদি। ইতোমধ্যে আমরা যা জেনেছি তার দ্বারা এটাই নির্দেশ করে যে প্রতিটি ভিন্ন তথ্য একেকটি আলাদা ভেরিয়বলে সংরক্ষণ করতে হবে। যেমন:

Dim EName as String, Dim EAddress as String, Dim ETelephone as String, Dim EAge as Integer, Dim ESalary as Decimal, Dim EJobTitle as String

এখন অপর আরেক কর্মচারীর জন্য অনুরূপভাবে ভেরিয়েবল ডিক্লারেশন দরকার। অন্য আরেক জনের জন্যও এরূপ করতে হবে। সুতরাং দশজন কর্মচারী হলে ৬০ টি ভেরিয়েবল লাগবে। বেশি হলে আরো বেশি লাগবে। এভাবে প্রোগ্রামিং করলে আপনি শীঘ্রই ইস্তিফা দেবেন! আসলে এরূপ করার দরকার নেই। আপনি Structure ব্যবহার করে একবার মাত্র ভেরিয়েবল ডিক্লারেশ করে কাজ সারতে পারেন। কিভাবে? এই দেখুন:

Structure ডিক্লারেশন

বিভিন্ন প্রোপার্টি একত্রে একটি Structure-এ এভাবে ডিক্লারেশন করা যায়:

```
Structure EmployeeInfo
    Dim EName as String
    Dim EAddress as String
    Dim ETelephone as String
    Dim EAge as Integer
    Dim ESalary as Decimal 'decimal number data type
    Dim EJobTitle as String
End Structure
```

এখন আমরা নতুন একটি ডাটা টাইপ তৈরি করলাম যার নাম EmployeeInfo। এর ভেতর আছে ৬টি ভিন্ন তথ্য। এবার EmployeeInfo-কে আমরা ডাটা টাইপ হিসাবে ডিক্লায়ার করতে পারি যেমনটি পারি String, Integer ইত্যাদি। সুতরাং আমরা ডিক্লায়ার করবো এভাবে:

```
Dim A as EmployeeInfo
A.EName = "Muhammad Abdus Samad"
A. EAddreaa ="10 Bashundara R/A, Maulvi Bazar, Bangladesh"
A.ETelephone ="01235 215226"
A.EAge = 35
A.ESalary = 32500.00
A.EJobTitle = "Managing Director"
```

স্টোর করা তথ্য আনতে হবে এভাবে: A.ETelephone, A.ESalary = 32500.00 ইত্যাদি। এ হলো কর্মচারী A। আমরা একই ভেরিয়েব টাইপ ব্যবহার করে আরো কর্মচারীর তথ্য সংরক্ষণ করতে পারি। যেমন:

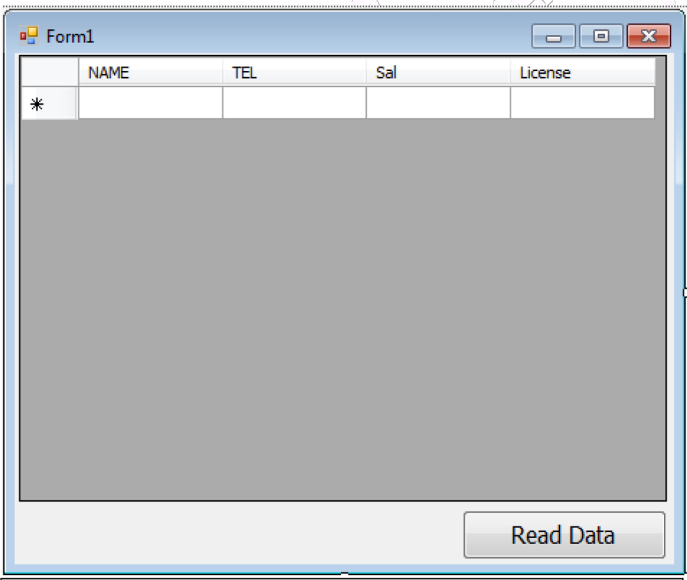
৬ষ্ট পরিচ্ছেদ: ***Structure***

বিভিন্ন ভেরিয়েবল্ একত্রে : *Different Variables together*

```
Dim B as EmployeeInfo
B.EName = "Abdur Rahim"
B. EAddreaa ="99 Sunamgonj Rd, Patantula R/A, Sylhet, Bangladesh"
B.ETelephone ="01723658211"
B.EAge = 33
B.ESalary = 22500.00
B.EJobTitle = "Chief Supervisor"
```

অনুরূপ আমরা যতোজন কর্মচারীর প্রয়োজন ততোজনের তথ্য EmployeeInfo এর মধ্যে সংরক্ষণ করতে পারি। সুতরাং বুঝাই যাচ্ছে Structure একটি গুরুত্বপূর্ণ ডিক্লারেশন।
এবার আমরা Structure ব্যবহারের একটি উদাহরণ দেখাবো। নিচের পদক্ষেপগুলো অনুসরণ করুন।

১. নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে যোগ করুন ১টি বোতাম ও একটি ডাটাগ্রিডভিউ [DataGridView]। গ্রিডে থাকবে: Name, Tel, Sal, License -এ চারটি কলাম। দেখুন নিচের চিত্রটি।



চিত্র -২৪: স্ট্রাকচার প্রজেক্টের ফর্ম

২. ফর্মের যে কোনো স্থানে ডবল ক্লিক করুন। কোড জানালা আসবে। এতে লিখুন নিচের কোডটি:

```
Public Class Form1
    'declare the structure
    Structure PersonInfo
        'person name
        Dim Name As String  'person telephone number
        Dim Tel As String
      'person salary
        Dim Sal As Decimal
      'person license number
        Dim LincenseNumber As String
    End Structure
' this is the array to store persons' info
   Dim Info() As PersonInfo
End Class
```

এ কোড ব্লক দ্বারা আমরা PersonInfo নামে একটি নতুন ভেরিয়েবল ডিক্লায়ার করেছি। এ ব্লকটি ফর্মের ক্লাস ডিক্লারেশন পরেই থাকবে। আলাদা কোনো সাবরুটিনের ভেতর একে ডিফাইন করা যাবে না। PersonInfo হলো একটি Stucture ভেরিয়েবল। এ ভেরিয়েবলের ভেতর কয়েকটি ভিন্ন ভেরিয়েবল ডিক্লারেশন আছে। এ ক্ষেত্রে এগুলো হলো: Name, Tel, Sal এবং LicenseNumber। লক্ষ করুন, এসব ভেরিয়বলের তিনটি String Type ও একটি Decimal Type ভেরিয়বল ডিক্লারেশ করা হয়েছে। স্ট্রাকচারের মধ্যে এভাবে ভেরিয়েবল ডিক্লারেশনের ফলে এটা সম্ভব হয়েছে। প্রোগ্রামের সর্বত্র পাওয়ার নিশ্চয়তার জন্য Info() নামক একটি PersonInfo টাইপ এ্যারে ভেরিয়েবলও এখানে ডিক্লায়ার করেছি। এই এ্যারের মধ্যে ব্যক্তির তথ্যাদি সংরক্ষণ করা হবে। এখন আমাদেরকে PersonInfo নামক ভেরিয়েবলে তথ্য ঢুকাতে হবে। এজন্য আমরা Read Data বোতামটি ব্যবহার করবো। এর ক্লিক ইভেন্ট হ্যান্ডলারে আমরা পরের পৃষ্ঠার কোডটি লিখবো:

৬ষ্ট পরিচ্ছেদ: ***Structure***

বিভিন্ন ভেরিয়েবল্ একত্রে : *Different Variables together*

```
' read the information and store it in an array
Public Sub ReadInfo()
    Dim I As Integer 'I for For-Next Loop
    Dim N As Integer 'N for number of persons
    On Error GoTo ChkNum ' incase the user enters a wrong entry [not number]
    N = InputBox("Enter the number of people") 'first enter how many person
    ReDim Info(0 To N - 1) 'redimension the array to 0 to number of person
    ' read info here
    For I = 0 To N - 1
        Info(I).Name = InputBox("Enter the name of person")
        Info(I).Tel = InputBox("Enter the telephone number")
        Info(I).Sal = InputBox("Enter the salary")
        Info(I).LincenseNumber = InputBox("Enter license number")
    Next
    Exit Sub
ChkNum: 'label jump
    Exit Sub
End Sub
```

উক্ত সাবটি কিভাবে কাজ করে তা আপনার মোটামুটি জানা হয়ে গেছে আশারাখি। এরপরও আবার বুঝিয়ে দিচ্ছি। প্রথমত সাবরুটিনটি নিজের বানানো মনে রাখবেন। আপনাকে সবকিছু নিজেই টাইপ করতে হবে। প্রথম লাইন দ্বারা ইন্টেজার ভেরিয়েবল ডিক্লারেশ করা হয়েছে। এই ইন্টেজার ফর-নেস্ট লুপের গণক হিসাবে ব্যবহৃত হয়েছে। দ্বিতীয় লাইনে ঘোষিত ইন্টেজার হলো লোকসংখ্যা। এটার মাত্রা চতুর্থ লাইনে নিশ্চিত হবে। ব্যবহারকারী যতো ঢুবাকেন তা-ই হবে এর মাত্রা। তৃতীয় লাইনটিকে বলে Error Handling। ভুলবশত ব্যবহারকারী সংখ্যা না ঢুকিয়ে অন্য কিছু ঢুকালে এই লাইন নিশ্চিত করবে সাব থেকে বের হতে। অন্যথায় প্রোগ্রাম ক্রাশ করবে। এই লাইন লেভেলে জাম্প করবে।

এবার সংগ্রহিত তথ্যাদি ডাটা-গ্রিড ভিউ-এ প্রদর্শনের প্রয়োজন। এজন্য আমরা নিজে একটি সাবরুটিন বানিয়েছি। এ'টি ফাংশনের মতো কাজ করবে। যখনই আমরা একে কল করবো তখনই এটা সক্রিয় হবে।

```
' fill the data grid with array info
Public Sub FillDGV(ByVal DAT() As PersonInfo, ByVal DGV As DataGridView)
    DGV.Rows.Clear()
    Dim I As Integer
    For I = 0 To DAT.Length - 1
        On Error GoTo endSub 'error handling. Jump to endSub label and exit
        DGV.Rows.Add(DAT(I).Name, DAT(I).Tel, DAT(I).Sal, DAT(I).LincenseNumber)
    Next
    Exit Sub
endSub: 'label jump
    MsgBox("Enter a whole number please!")
    Exit Sub
End Sub
```

উপরের সাব কিভাবে কাজ করে তা একটু তলিয়ে দেখা যাক। প্রথমত জেনে রাখি, এই সাবটি কল করা হবে নিচের দু'টি বোতামের সাব থেকে। কল করলেই এটা কাজ শুরু করবে। প্রথম লাইন তথা DGV.Rows.Clear() সকল রো ক্লিয়ার করবে। দ্বিতীয় লাইনে আমরা ফর-নেস্ট লুপের জন্য একটি কন্ট্রোল ইন্টেজার ডিক্লায়ার করেছি। লক্ষ করুন: ক'বার ফর-নেস্ট লুপ ঘুরবে তা নির্ভর করে ঠিক ক'জন ব্যক্তির তথ্য আমাদের এ্যারের মধ্যে সংরক্ষিত আছে- তার ওপর। লুপের লাইনটিতে DAT(I).Name, DAT(I).Tel, DAT(I).Sal, DAT(I).LincenseNumber দ্বারা একে একে সবগুলো রো পূর্ণ হবে। এবার দেখে নিই বাকি কোডগুলো।

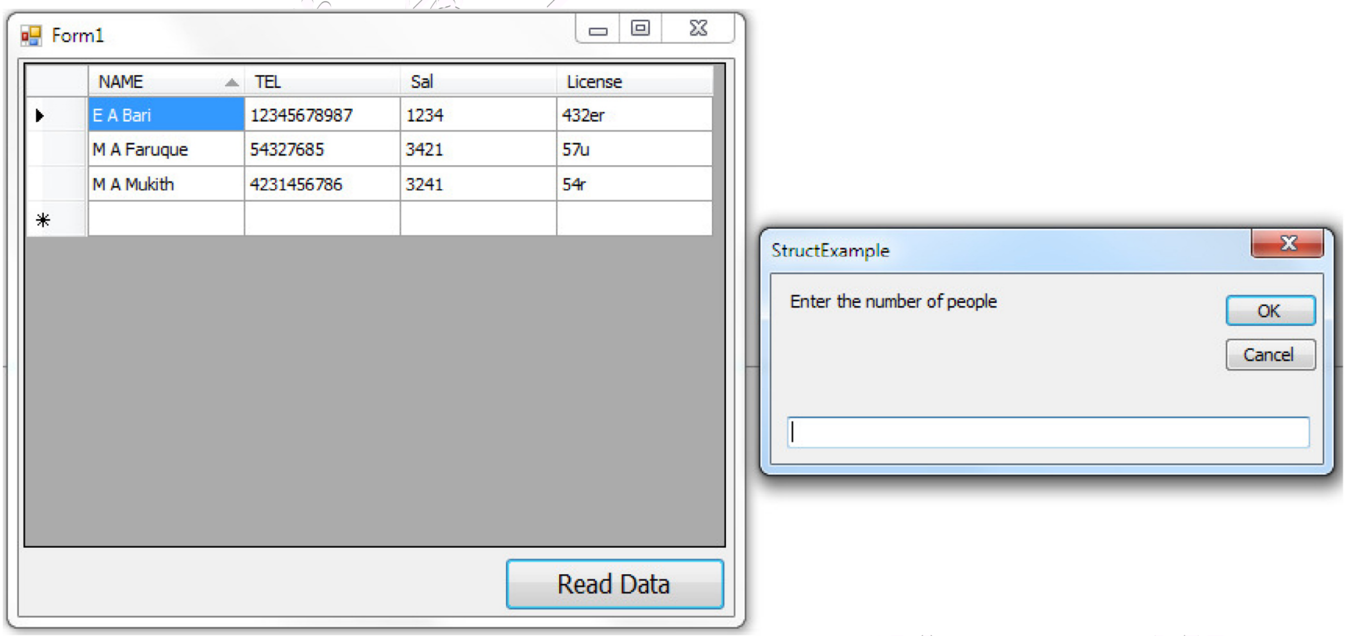
৬ষ্ট পরিচ্ছেদ: ***Structure***

বিভিন্ন ভেরিয়েবল্ একত্রে : *Different Variables together*

Read Data বোতাম দ্বারা আমরা ব্যক্তিদের তথ্য সংগ্রহ করবো। এই বোতামের ক্লিক ইভেন্টে নিচের কোডটি লিখতে হবে। এই বোতাম দ্বারা ব্যবহারকারী প্রয়োজন পরিমাণ তথ্য ঢুকানোর পর গ্রিডে সবকিছু প্রদর্শিত হবে।

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
    ReadInfo() 'call sub to enter data by user
    FillDGV(Info, DataGridView1) 'call sub to display data in the grid
End Sub
```

লক্ষ করুণ এই সাবটির মধ্যে মাত্র দু'টি লাইন আছে। উভয় লাইন দ্বারাই আমরা ইতোমধ্যে ব্যাখ্যাত নিজের তৈরি সাব কল করছি। প্রথমে আমার কল করেছি ReadInfo() সবাকে। এটা ব্যবহারকারীকে তথ্য ঢুকানোর জন্য ডায়লগ প্রদর্শন করবে। তথ্য ঢুকানো শেষ হলেই দ্বিতীয় সাব FillDGV(Info, DataGridView1) কল করা হবে। এ সাবের কাজ হলো তথ্যাদি গ্রিডে প্রদর্শন করা। ব্যস! স্ট্রাকচার ব্যবহারের উদাহরণ প্রোগ্রাম শেষ হলো। এবার চালু করে দেখার পালা। নিচের চিত্রটি দেখুন। লক্ষ করুন, NAME-এ ক্লিক করলে সমগ্র তথ্য আক্ষরিকভাবে সোর্ট হবে।



চিত্র -২৫: স্ট্রাকচার প্রোগ্রাম চলাকালীন চিত্র

পরের পরিচ্ছেদে আমরা ভিজুয়্যাল বেসিক.নেট এর আরো একটি গুরুত্বপূর্ণ বিষয়ের উপর উদাহরণসহ আলোচনা করবো। এ বিষয়টি হলো অন্যত্র কোড লিখে সংরক্ষণ ব্যবস্থা। এ পর্যন্ত আমরা যা করেছি তা সবই ফর্মের ক্লাস সাবরুটিনের ভেতর ছিলো। এখন একটি কোড ব্লকের উপর আলোচনা করবো যা লিখা হবে বাইরে- আলাদাভাবে। তবে অবশ্যই এটা যে কোনো জায়গায় ব্যবহারযোগ্য হবে। এই ব্যবস্থার নাম মডিউল [Module]।

৭ম পরিচ্ছেদ: *Separate code block in Program*

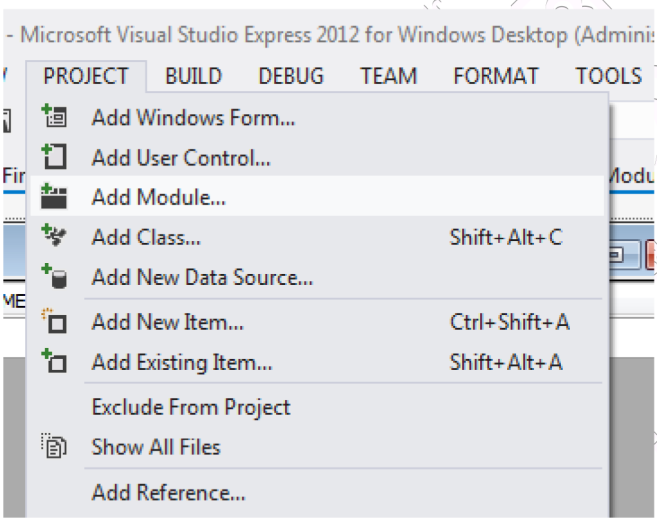
মডিউল ও ক্লাস : *Module & Class*

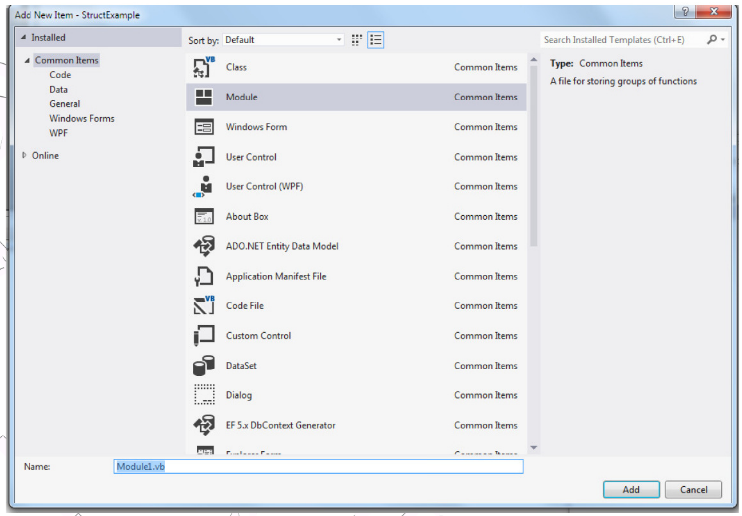
ভিজুয়্যাল বেসিক.নেটে আপনি আলাদাভাবে কোড ফাইল লিখে যে কোনো স্থানে সংরক্ষণ করতে পারেন। এটা খুব সুবিধাজনকও বটে। আপনি একই কোড বিভিন্ন প্রোগ্রামে ব্যবহার করতে চাইলে এ সুবিধা থাকাটা খুব জরুরীও।

মডিউল [Module]

মডিউলে শুধুমাত্র কোড থাকে। অর্থাৎ ফাংশন, সাব, স্ট্রাকচার ইত্যাদি ছাড়া আর কিছুই থাকবে না। এতে না আছে কোনো উইন্ডো, ফর্ম, বোতাম কিংবা গ্রাফিক্যাল ইউজার ইন্টারফেইস [GUI]। মডিউলে থাকবে শুধুমাত্র কোড।

প্রজেক্টে মডিউল যুক্ত করতে হলে প্রজেক্ট [Project] মেন্যুতে যেয়ে এড মডিউল [Add Module] চয়েজ করতে হবে। দেখুন নিচের ছবিটি (বায়ে)। Add Module চয়েজ করলে আসবে আরেকটি জানালা। ডানের চিত্রটি দেখুন।





এভাবে নতুন মডিউল বানিয়ে প্রজেক্টে যুক্ত করার পর মডিউলের কোড জানালা আসবে। আপনি এতে সাব, ফাংশন, স্ট্রাকচার, ইত্যাদি ডিক্লায়ার করতে পারেন। এসব বস্তু প্রোগ্রামের যে কোনো স্থান থেকে কল করে ব্যবহার করতে পারেন। সর্বোপরি এই মডিউল অন্য কোনো প্রজেক্টেও নিতে পারেন। আর মিডউল শব্দের অর্থই এটা। নিচে মডিউলের উদাহরণ কোড দেখিয়েছি। এর সাব ও ফাংশন কিংবা যে কোনো একটি প্রোগ্রামের অন্যত্র থেকে কল করতে পারেন। একটু স্টাডি করে নিন।

```
Public Module ThisModule
   Sub Main()
      Dim userName As String = InputBox("What is your name?") 'ask for user name
      MsgBox("User name is: " & userName) 'show it in message box
   End Sub
   Function Sq(SupNum As Double) As Double 'returns square of number
      Return SupNum * SupNum
   End Function
   ' Insert variable, property, procedure, and event declarations.
End Module
```

মডিউল ও ক্লাসের ব্যবহার এ পরিচ্ছেদের শেষে একটি উদাহরণে দেখিয়েছি।

৭ম পরিচ্ছেদ: *Separate code block in Program*

মডিউল ও ক্লাস : *Module & Class*

ভিজুয়্যাল বেসিক.নেট -এ মডিউলের বৈশিষ্ট্যাবলী:
১. এগুলো ভিজুয়্যাল বেসিক ফাইল।
২. এগুলোর মধ্যে প্রোগ্রামের সর্বত্র ব্যবহারযোগ্য সাব, ফাংশন ইত্যাদি সংরক্ষণ করা যায়।
৩. এগুলো ব্যবহার করলে বড় প্রোগ্রাম নিয়ন্ত্রণ ও রক্ষণাবেক্ষণ অনেকটা সহজ হয়।
৪. প্রোগ্রাম বুঝা সহজ হয়।
৫. এগুলো প্রোগ্রামে থাকলেও ব্যবহার করা ও না করার এখতিয়ার আছে।
৬. একাধিক মডিউল যে কোনো প্রোগ্রামে ব্যবহার করা যেতে পারে।
৭. মডিউল অপর কোনো প্রোগ্রামে নেওয়া যেতে পারে।

আমরা এবার আরো একটি একক কোড উইনিট নিয়ে আলোচনা করবো। এটা অনেকের মতে মডিউল থেকেও উন্নতমানের। এই বস্তুর নাম ক্লাস [Class]।

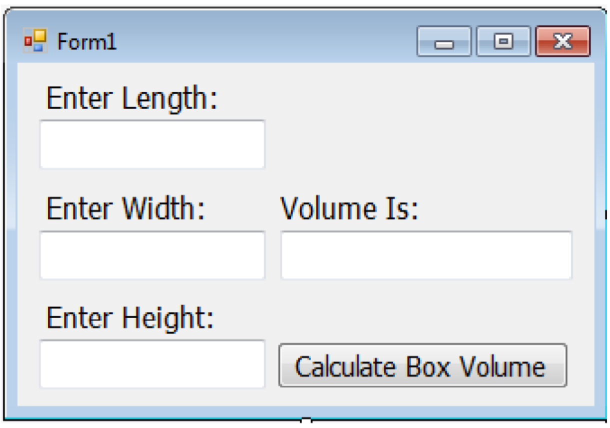
ক্লাস [Class]

ক্লাস অনেকটা স্ট্রাকচারের মতো কাজ করে। অতিরিক্ত সুযোগ হলো এতে সাবরুটিন ও ফাংশন যুক্ত করা যায়। ক্লাসের কোড পুনর্ব্যবহার করাও সম্ভব। এই ক্লাসের ভেতরই উইন্ডোজ ফর্ম প্রোগ্রামের কোড লিখিত থাকে। যেমন ফর্মের ক্লাস ডিক্লারেশন থাকে নিম্নরূপ:

```
Public Class Form1
   'Here all the subs and functions
End Class
```

অনুরূপ আমরা নিজেই আরো ক্লাস ডিক্লারেশন করতে পারি। নতুন ক্লাস বানানোর পদ্ধতি নিম্নরূপ:
১. প্রথমে প্রজেক্ট [Project] মেন্যুতে যেয়ে এড ক্লাস [Add Class] চয়েজ করুন।
২. ক্লাসের নাম দিন।
৩. সংরক্ষণ করুন। ক্লাস মডিউলের মতো আলাদা ফাইলে সংরক্ষিত হবে।

নিজের তৈরি ক্লাসের ব্যবহার সঠিকভাবে বুঝার জন্য এবার আমরা একটি প্রজেক্ট তৈরি করবো। আমাদের এ প্রোগ্রামের উদ্দেশ্য হলো একটি বাক্সের ভলিউম বের করে প্রদর্শন করা। এই প্রোগ্রাম যদিও অন্যভাবে সহজে লেখা যাবে তথাপি আমরা এটা একটি মডিউলের ভেতর ক্লাস ডিক্লারেশন দ্বারা করতে চাই। নিচের পদক্ষেপগুলো একে একে অনুসরণ করে প্রজেক্ট সৃষ্টি করুন ও কোড লিখুন।
১. নতুন প্রজেক্ট তৈরি করুন। নামকরণ করুন ModClassExample।
২. ফর্মের মধ্যে যুক্ত করুন ৪টি টেক্সট্ বাক্স ও একটি বোতাম। প্রত্যেক টেক্সট্ বাক্সের উপরে যুক্ত করুন একেকটি লেবেল। ডানের চিত্রের মতো পুরো ফর্মটি ডিজাইন করুন। ফর্মের FormBorderStyle হবে Fixed3D এবং MaximizeBox হবে False।
৩. প্রত্যেক লেবেল ও টেক্সট্ বাক্সের ফন্ট সাইজ হবে ১২।
৪. ডানের টেক্সট্ বাক্সের প্রপার্টিতে যেয়ে চয়েজ করুন: ReadOnly [True]। ব্যস! এ বাক্সে আর কিছু লেখা যাবে না। কারণ আমরা এতে শুধুমাত্র ফলাফল প্রদর্শন করবো।
৫. TextBox1, TextBox2 ও TextBox3 এর KeyPress Event সাবে [পরের পৃষ্ঠার] এই কোড লাইনগুলো যথাক্রমে লিখে দিন। এর ফলে এসব বাক্সে সংখ্যা ছাড়া আর কিছুই লিখা সম্ভব হবে না।



৭ম পরিচ্ছেদ: *Separate code block in Program*

মডিউল ও ক্লাস : *Module & Class*

```
'this three subs makes sure only decimal numbers are entered into textBox1, TextBox2 & TextBox3
Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress
e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
 End Sub
 Private Sub TextBox2_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox2.KeyPress
e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
 End Sub
 Private Sub TextBox3_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox3.KeyPress
e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
 End Sub
```

এবার প্রোগ্রামটি রান করে দেখুন। প্রথম তিনটি বাক্সে সংখ্যা ছাড়া অন্য কিছু লিখার চেষ্টা করুন। পারবেন না। তবে সংখ্যা [দশমিক বিন্দুসহ] ঠিকই লিখতে পারবেন। উপরের কোডগুলো গুরুত্বপূর্ণ। সংখ্যা ছাড়া অন্য কিছু থাকলে প্রোগ্রাম ক্রাশ করবে- যা আমাদের ইচ্ছে নেই।

৬. প্রোগ্রাম বন্ধ করে প্রজেক্ট মেন্যু থেকে চয়েজ করুন Add Class। ক্লাসের নামকরণ করুন: ClassBox.vb।

৭. ক্লাসের কোড হবে নিম্নরূপ:

```
Public Class ClassBox
   Public length As Double   ' Length of a box
   Public width As Double   ' Width of a box
   Public height As Double   ' Height of a box
End Class
```

৭. এবার প্রজেক্ট মেন্যু থেকে চয়েজ করুন Add Module। মডিউলের নামকরণ করুন: BoxMod.vb। কোড হবে:

```
Module BoxMod
  'define Volume Function here / public function available anywhere in the programme
  Public Function Vol(SupVL As Double, SupVW As Double, SupVH As Double) As Double
     Return SupVL * SupVW * SupVH
  End Function
End Module
```

৮. ডবল ক্লিক করুন Calculate Box Volume বোতামে। প্রদর্শিত সাবে লিখুন:

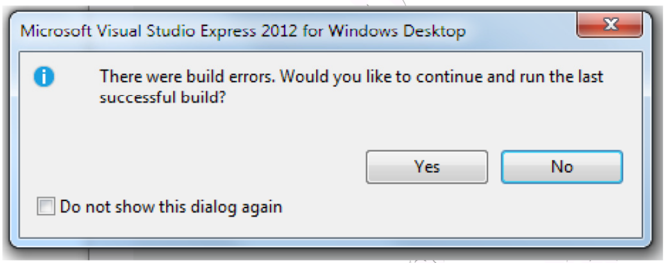
```
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      'make sure user entered numbers
      If TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Then
         Exit Sub
      End If
      Dim Cl As New ClassBox 'declare a class type
      Cl.length = TextBox1.Text 'get value
      Cl.width = TextBox2.Text 'get value
      Cl.height = TextBox3.Text 'get value
    'Make sure they are numbers - in case user enters just space. Although in our example space not possible
     If IsNumeric(Cl.length) = False Or IsNumeric(Cl.width) = False Or IsNumeric(Cl.height) = False Then
         Exit Sub
      End If
      Dim ResVal As Double = Vol(Cl.length, Cl.width, Cl.height)
      TextBox4.Text = ResVal
   End Sub
```

৭ম পরিচ্ছেদ: *Separate code block in Program*

মডিউল ও ক্লাস : ***Module & Class***

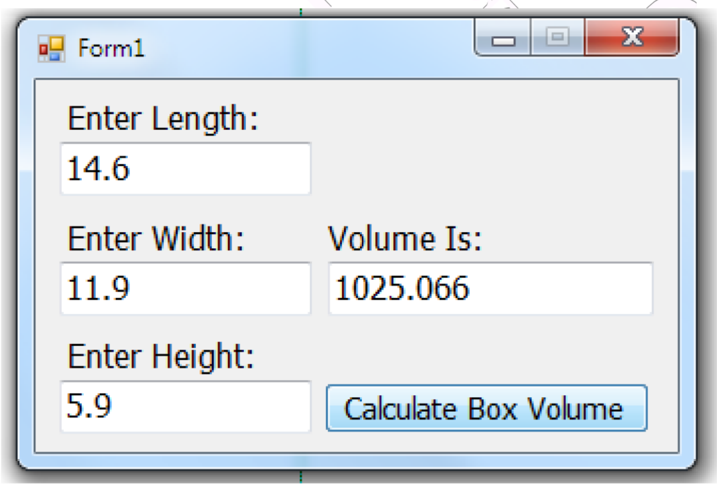
Compile and Test

ইতোমধ্যে আমরা প্রজেক্ট Start করে বার বার প্রোগ্রাম চালিয়ে পরীক্ষা করেছি। বাস্তবে Start মেন্যুটি চয়েজ করার পর কী হয়? সকল কোড প্রথমে এক্সপ্রেস ২০১২ বাইনারি কোডে পরিণত করে- একে বলে Compile। বাইনারি কোড মানে ০ এবং ১। আর ০ মানে সুইচ অফ এবং ১ মানে সুইচ অন। কম্পাইলার প্রথমে কোডের প্রতিটি লাইন একে একে পরীক্ষা করে দেখে এতে কোনো ভুল বা সিনটেক্সট এ্যারোর আছে কি না। যদি থাকে তাহলে কম্পাইলের পূর্বে একটি ওয়ার্নিং ডায়লগ আসবে। ডায়লগটি নিম্নরূপ:



সাধারণত আপনি ‘No’ বোতামে ক্লিক করবেন। তখন ভুলটি কোথায় হয়েছে তা-ও দেখাবে। ঠিক করার পর আবার Start Menu চয়েজ করে কম্পাইল করবেন। আপনার প্রোগ্রাম সম্পূর্ণ হয়ে গেলে কম্পাইল করে একটি **.exe** ফাইল তৈরি করবেন। এই কম্পাইল করা ফাইলই হলো এ্যাপলিকেশনের সর্বশেষ রূপ। একে পরিবর্তন করার কোনো সুযোগ আর থাকবে না। অন্যকথায় ডিস্ট্রিবিউশন পরে ব্যবহারকারী এই ফাইলে ডবল ক্লিক করে প্রোগ্রাম চালাবে। ব্যস! ভুল ত্রুটি সংশোধন একমাত্র আপনার কাজ। একে বলে ডিবাগিং। আপনি পরবর্তী ভার্শনে সংশোধিত আরেকটি **.exe** ফাইল [প্রয়োজনে Installer-সহ] পুনঃরিলিজ করবেন।

আপনার প্রোগ্রাম যদি বেশি জটিল হয় তাহলে বিল্ড মেন্যু [BUILD] চয়েজ করে আপনি ‘Publish’ [আপনার প্রেগ্রামের নাম]’ চয়েজ করবেন। এরপর প্রদর্শিত নির্দেশনাবলী মুতাবিক পাবলিশ করে সংরক্ষণ করবেন। যাক, এ ব্যাপারে গ্রন্থের শেষের দিকে আরো বিস্তারিত আলোচনা হবে। এখন আমরা এইমাত্র সম্পন্ন ModClassExample প্রোগ্রামটি Start করে টেস্ট করবো। দেখুন নিচের চিত্রটি।



পরের পরিচ্ছেদে আমরা Sub এবং Function নিয়ে উদাহরণসহ অতিরিক্ত কিছু আলোচনার প্রয়াস পাচ্ছি। এর কারণ হলো, সকল ভিজুয়্যাল বেসিক.নেট মূলত এ দু’টোকে নিয়েই তৈরি হয়। সুতরাং এগুলো সম্পর্কে যা-ই আমরা জানবো তা-ই লাভজনক হবে।

৮ম পরিচ্ছেদ: *More on Subs & Functions*

সাব ও ফাংশন : *Subs & Functions*

আমরা ইতোমধ্যে যাকিছু কোডিং করেছি তা সবই ছিলো মূলত সাব ও ফাংশন। হ্যাঁ, আমরা মডিউল এবং ক্লাস সম্পর্কেও জেনেছি। কিন্তু এগুলো আলাদা কোড ব্লক ছাড়া আর কিছু নয়। এগুলোর ভেতরও থাকে সাব এবং ফাংশন। সুতরাং সাব ও ফাংশন সম্পর্কে বেশি জানার অর্থ হলো .নেট কোডিং সম্পর্কে আরো বেশি অবগত হওয়া।

Writing Sub Procedure -সাব প্রসিডিউর লেখা

টেকনিক্যালী সাবরুটিনকে সাব প্রসিডিউর বলে। প্রত্যেক সাব প্রসিডিউর কল করা হয় তার নাম দিয়ে। এ কল হতে পারে অন্য কোনো ইভেন্ট সাব প্রসিডিউর থেকে কিংবা অন্য কোনো ইউজার-ডিফাইন্ড [অর্থাৎ আপনার বানানো] সাব প্রসিডিউর থেকে। এরা এক বা একাধিক তথ্য অথবা আর্গুমেন্ট গ্রহণ করে এর ওপর কাজ করতে পারে। কাজ থেকে ফলাফল ফেরৎ পাঠাতেও পারে। কিন্তু ফাংশনের মতো নামের সাথে ভ্যালু বা মান রিটার্ন করতে পারে না। অন্যকথায় সাবরুটিন ব্যবহৃত হয়: ১. ইনপুট (প্রদত্ত ভ্যালু) গ্রহণ বা প্রসেস করতে, ২. আউটপুট (ফলাফল) প্রদর্শন করতে ও ৩. প্রোপার্টিজ [বৈশিষ্ট্যাবলী] সেট করতে।

এবার একটি উদাহরণ দেখবো। সাব প্রসিডিউর প্রায়ই ব্যবহৃত হয় ইনপুট নিয়ন্ত্রণে। এই উদাহরণে আমরা 'AddName' নামক একটি Sub Procedure তৈরি করবো। এটা ব্যবহারকারীকে লেখা প্রবেশের জন্য অনুরোধ জানাবে এবং তা টেক্সট্ বাক্সে প্রদর্শন করে দেখাবে। প্রসিডিউরটি একটি মডিউলে থাকবে। ফলে একে একাধিকবার কল করা যাবে। মূলত পুরো প্রেগ্রামের যে কোনো স্থান থেকে একে কল করে ব্যবহার করা যেতে পারে।

১. একটি নতুন প্রজেক্ট তৈরি করুন। এর নামকরণ করুন: 'My Text Box Sub'।
২. ফর্মে দু'টি টেক্সট্ বক্স যুক্ত করুন।
৩. বাক্স দু'টোর উপরে দু'টি লেবেল যুক্ত করুন। এগুলোতে থাকবে ডিপার্টমেন্টের নাম।
৪. মোট তিনটি বোতাম যুক্ত করুন। প্রথম দু'টি যথাক্রমে টেক্সট্ বাক্স ১ ও টেক্সট্ বাক্স ২ এর নিচে রাখুন। বাকিটি রাখুর ফর্মের নিচের দিকে।
৫. নিচের টেবিল মুতাবিক বিভিন্ন বস্তুর প্রোপার্টি সেট করুন:

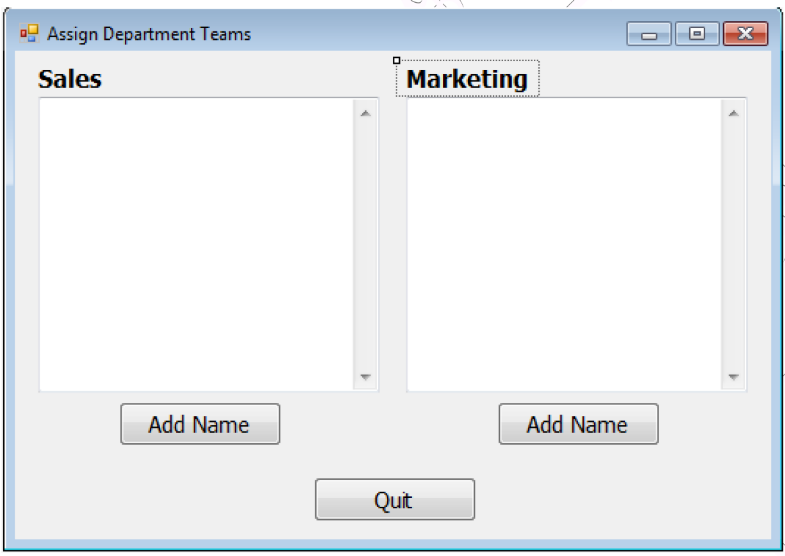
বস্তু	প্রোপার্টি	সেটিং
TextBox1	Multiline	True
	Name	txtSale
	ReadOnly	True
	ScrollBars	Vertical
	TabStop	False
	Font Size	12
TextBox2	All Same accept Name txtMkt	
Label1	Font	Bold
	Font Size	12
	Name	lblSales
	Text	"Sales"
Label1	Font	Bold
	Size Size	12
	Name	lblMkt
	Text	"Marketing"

৮ম পরিচ্ছেদ: *More on Subs & Functions*

সাব ও ফাংশন : *Subs & Functions*

বস্তু	প্রোপার্টি	সেটিং
Button1	Name	btnSales
	Text	"Add Name"
	Font Size	11
Button2	Name	btnMkt
	Text	"Add Name"
	Font Size	11
Button3	Name	btnQuit
	Text	"Quit"
	Font Size	11
Form1	Text	"Assign Department Teams"

সব ঠিকমতো হলে ফর্মটির ডিজাইন হবে নিচের চিত্রের মতো।



এখন আমরা প্রজেক্টে একটি মডিউল যুক্ত করবো। আর এতেই লিখবো আমাদের সর্বত্র ব্যবহারযোগ্য AddName সাবরুটিনটি।

৬. প্রজেক্ট মেন্যু থেকে চয়েজ করুন Add Module। নতুন মডিউল তৈরি হবে। এটার নামকরণ যা আছে তা-ই রেখে দিন।

৭. মডিউল তৈরির পরই এর কোড জানালা খুলবে। এই কোড জানালায় থাকবে নিচের কোড ব্লকটি:

```
Module Module1
   'Our Sub here
End Module
```

আমরা যে সাব প্রসিডিউর লিখবো তা এই ব্লকের ভেতরই থাকবে।

৭. নিচের কোডটি হবে মডিউলের মধ্যে।

```
Module Module1
   Sub AddName(ByVal Team As String, ByRef ReturnString As String)
      Dim Prompt, Nm, WrapCharacter As String
      Prompt = "Enter a " & Team & " employee."
      Nm = InputBox(Prompt, "Input Box")
      WrapCharacter = Chr(13) + Chr(10)
      ReturnString = Nm & WrapCharacter
   End Sub
End Module
```

ব্যাখ্যা: এই প্রসিডিউর কল হলে দু'টি আর্গুমেন্ট গ্রহণ করবে। এগুলো হলো Team, স্ট্রিং টাইপ ডিপার্টমেন্টের নাম এবং ReturnString, একটি শূন্য স্ট্রিং ভেরিয়েবল যাতে থাকবে কর্মকতার নাম। লক্ষ্য করুন, ReturnString-কে

৮ম পরিচ্ছেদ: ***More on Subs & Functions***

সাব ও ফাংশন : *Subs & Functions*

ByRef কিওয়ার্ডসহ ডিক্লায়ার করা হয়েছে। এর ফলে প্রসিডিউরে এই আর্গুমেন্টে কোনো পরিবর্তন হলে তা কলিং রুটিনে ফিরে দেওয়া হবে। প্রথম লাইনে আমরা তিনটি স্ট্রিং ভেরিয়েবল ডিক্লায়ার করেছি। এগুলো ২য়, ৩য় ও ৪র্থ লাইনে ইনিশিয়েলাইজ করা হয়েছে। শেষের লাইনে রিটার্নস্ট্রিংকে ফর্মেট করেছি। ব্যস! এটুকুই। এই প্রসিডিউর ব্যবহারকালে বুঝতে পারবেন এটা কিভাবে কাজ করে।

৮. পুনরায় ফর্ম ডিজাইন প্রদর্শন করুন। ডবল ক্লিক করুন ফর্মের Add Name বোতামে। বোতামের ক্লিক ইভেন্ট প্রসিডিউর খুলবে। এতে যুক্ত করুন নিচের কোডটি।

```
'calls AddName Procedure defined in Module1
Private Sub btnSales_Click(sender As Object, e As EventArgs) Handles btnSales.Click
    Dim SalesPosition As String = ""
    AddName("Sales", SalesPosition)
    txtSale.Text = txtSale.Text & SalesPosition
End Sub
```

ব্যাখ্যা: AddName প্রসিডিউর কলে দু'টি আর্গুমেন্ট আছে: Sales এবং SalesPosition। শেষের লাইনের কাজ হলো রেফারেন্সের মাধ্যমে প্রেরিত আর্গুমেন্ট ব্যবহার করে txtSale টেক্সট বাক্সে লেখা যুক্ত করা। কনক্যারেটশন অপারেটর & এর কারণে নতুন নাম টেক্সটের শেষে যুক্ত হবে। যাক, প্রোগ্রাম চলাকালে বিষয়টি আরো পরিষ্কার হবে।

৯. এবার ডবল ক্লিক করুন ডানের Add Name বোতামে। কোড এডিটরে বোতামের ক্লিক ইভেন্ট সাব দেখাবে। এই সাবের ভেতর লিখুন:

```
Private Sub btnMkt_Click(sender As Object, e As EventArgs) Handles btnMkt.Click
    Dim MktPosition As String = ""
    AddName("Marketing", MktPosition)
    txtMkt.Text = txtMkt.Text & MktPosition
End Sub
```

ব্যাখ্যা: এই বোতামের কোড প্রায় একই। পার্থক্য হলো মডিউলের AddName প্রসিডিউরে "Marketing" পাটিয়ে দেয় এবং আপডেট করে txtMkt টেক্সট বাক্স।

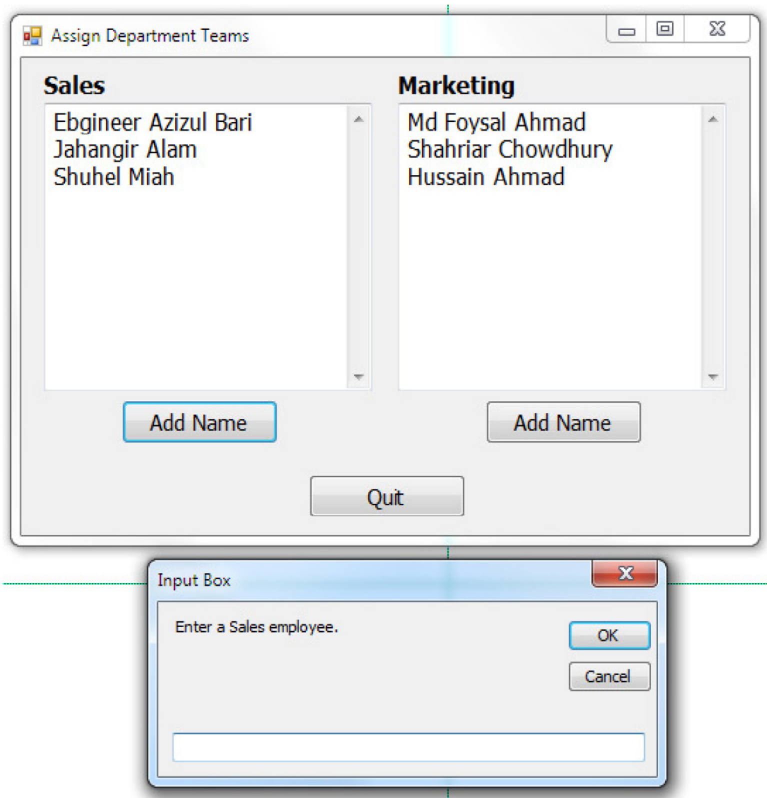
১০. এবার ডবল ক্লিক করুন Quit বোতামে। এর ক্লিক ইভেন্টে লিখুন:

```
'This sub quits the application
Private Sub btnQuit_Click(sender As Object, e As EventArgs) Handles btnQuit.Click
    End
End Sub
```

১১. এবার Start মেন্যুতে ক্লিক করে প্রোগ্রাম পরিচালনা করুন। উভয় Add Name বোতামে ক্লিক করে দেখুন। পরের পৃষ্ঠায় প্রোগ্রাম চলাকালীন চিত্র দেখিয়েছি। এ উদাহরণ দ্বারা এটাই দেখানো হলো যে, একটি মাত্র সাব প্রসিডিউর ব্যবহার করে দুই বা ততোধিক ইনপুট কাজ করা যায়। এই মৌলিক টেকনিক কাজে লাগিয়ে বার বার ব্যবহারযোগ্য শক্তিশালী সাব এবং (পরবর্তীতে আলোচ্য) ফাংশন সৃষ্টি করতে পারেন আপনার প্রোগ্রামের জন্য।

৮ম পরিচ্ছেদ: ***More on Subs & Functions***

সাব ও ফাংশন : *Subs & Functions*



প্রত্যেকবার Add Name বোতামে টিপ দিলেই নিচের ইনপুট বক্স ডায়লগ আসবে এবং যে নাম লিখবেন তা টেক্সট বাক্সে প্রদর্শিতও হবে। সুতরাং আপনি কোম্পানির সেল্স ও মার্কেটিং কর্মকর্তাদের একটি নামের লিস্ট বানাতে সক্ষম হলেন। হয়তো ভাবছেন এটা সংরক্ষণ করা যায় কিভাবে? এ ব্যাপারে ফাইল ম্যানেজমেন্টের উপর আলোচনাকালে ব্যাখ্যা আসবে। আপাতত এটাই জানা উদ্দেশ্য যে, একই প্রসিডিউর বার বার ব্যবহার করা সম্ভব।

Writing Function Procedure - ফাংশন প্রসিডিউর লেখা

আমরা জানি ফাংশন প্রসিডিউর হলো একগুচ্ছ ভিজুয়্যাল বেসিক.নেট কোড যা লেখা হয় Function এবং End Function এর মাঝখানে। এই কোডগুলোর কাজ হলো কোনো অর্থপূর্ণ কর্ম সম্পাদন। যেমন: টেক্সট প্রসেস করা, ইনপুট নিয়ন্ত্রণ কিংবা কোনো মান বেরকরণ ইত্যাদি কাজে লাগাতে ফাংশন কল করতে হয়। ফাংশনে প্রয়োজনে আর্গুমেন্ট বা প্যারামিটার যুক্ত করা যায়। কলের সময় ব্র্যাকেটের ভেতর এসব প্যারামিটারের নাম দিতে হবে। মূলত কিম্পউটারে অসংখ্যা রেডি-মেইড ফাংশন প্রসিডিউর আছে। প্রত্যেক এ্যাপলিকেশনই প্রয়োজনমাফিক এসব ফাংশন ব্যবহার করে থাকে। আমরাও ব্যবহার করেছি। যেমন: MsgBox, InputBox ইত্যাদি। নিজের তৈরি ফাংশন ও এসব সিস্টেম ফাংশনে মূলত তেমন কোনো পার্থক্য নেই। ফাংশনের আরেকটি বৈশিষ্ট্য হলো এগুলো যদি আমরা আলাদা মডিউলে ডিফাইন করি তাহলে সমগ্র প্রোগ্রামব্যাপী তা ব্যবহার করতে পারি। একই কথা সাব প্রসিডিরের ক্ষেত্রেও সত্য।

ফাংশনের আরেকটি উদাহরণ উপস্থাপনের পূর্বে তিনটি ব্যাপার পরিষ্কার করা প্রয়োজন বোধ করছি। যে কোনো ফাংশনের সিনটেক্সটে যা থাকা জরুরী তার মধ্যে এ তিনটি গুরুত্বপূর্ণ: ১. ফাংশনের নাম, ২. রিটার্ন টাইপ (যদি না থাকে তাহলে ভিজুয়্যাল বেসিক 'অবজেক্ট' টাইপ রিটার্ন করবে] ও ৩. ফাংশন স্টেটম্যান্টস্। যেমন:

নিচের ফাংশন দু'টি টেক্সট মান হিসাব করে যোগফল রিটার্ন করে। রিটার্ন মান হবে সিঙ্গুল ডাটা টাইপ।

```
Function TotalTax(ByVal Cost as Single) As Single
    Dim StateTax, CityTax As Single
    StateTax = Cost * 0.05 'State tax is 5% 'value of cost comes from calling line
    CityTax = Cost * 0.015 'City tax is 1.5%
    TotalTax = StateTax + CityTax 'to return value function name is used.
    [Return StateTax + CityTax] 'রিটার্ন কিওয়ার্ড দ্বারাও মান ফেরৎ পাঠানো যায়।
End Function
```

কলিং: উক্ত ফাংশন কল করার উদাহরণ: TextBox1.Text = TotalText(400)।

লক্ষ করুন: ফাংশন নাম দ্বারা কিংবা **Return** কিওয়ার্ড দ্বারা মান রিটার্ন করা যায়। তবে উভয়টা ব্যবহার করা যাবে না।

যাক, এখন আমরা একটি নতুন প্রজেক্ট করবো এবং ফাংশনের ব্যবহার দেখাবো।

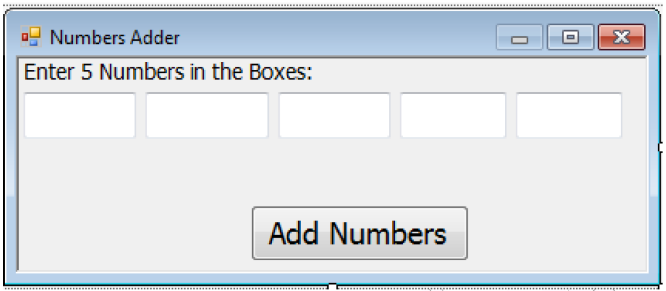
৮ম পরিচ্ছেদ: *More on Subs & Functions*

সাব ও ফাংশন : *Subs & Functions*

NumbersAdder উদাহরণ

ফাংশনের ব্যবহার দেখাতে যেয়ে আমরা একটি নতুন প্রজেক্ট তৈরি করবো। আমাদের উদ্দেশ্য হলো ব্যবহারকারী ৫টি সংখ্যা পাঁচটি টেক্সট্ বাক্সে লিখবেন। এগুলোর যোগফল একটি বোতামে ক্লিক করে বের করা।

১. একটি নতুন প্রজেক্ট তৈরি করুন। নামকরণ করুন: NumbersAdder।
২. ফর্মে থাকবে দু'টি লেবেল, ৫টি টেক্সট্ বাক্স ও ১টি বোতাম।
৩. ফর্ম ডিজাইন করুন নিচের ছবির মতো।



৪. ফর্মের Text প্রোপার্টি হবে Numbers Adder। FormBorderStyle প্রোপার্টি হবে Fixed3D এবং MaximizeBox প্রোপার্টি হবে False।
৫. label1 এর Text প্রোপার্টি হবে Enter 5 Numbers in the Boxes:।
৫. Button1 এর Text প্রোপার্টি হবে Add Numbers।
৬. উভয় লেবেলের ফন্ট সাইজ হবে ১০ এবং সকল টেক্সট্বাক্স ও বোতামের ফন্ট সাইজ হবে ১৪।

কোডিং

প্রথমত আমাদেরকে প্রতিটি টেক্সট্ বাক্সে যাতে দশমিক কিংবা আস্ত সংখ্যা ছাড়া আর কিছু লেখা যায় না তা নিশ্চিত করতে হবে। মনে আছে, ভেলিডেশন কোড? নিচের কোড লাইনটি প্রত্যেক টেক্সট্ বাক্সের KeyPress ইভেন্ট হ্যান্ডলারে লিখতে হবে:

```
e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
```

এবার নিচের ফাংশনটি ফর্মের ক্লাসের ভেতর লিখতে হবে:

```
'this function takes a list of numbers separated by commas and returns total of the numbers
Public Function calcSum(ByVal ParamArray args() As Double) As Double
    calcSum = 0
    If args.Length <= 0 Then Exit Function
    For i As Integer = 0 To UBound(args, 1)
        calcSum += args(i)
    Next i
End Function
```

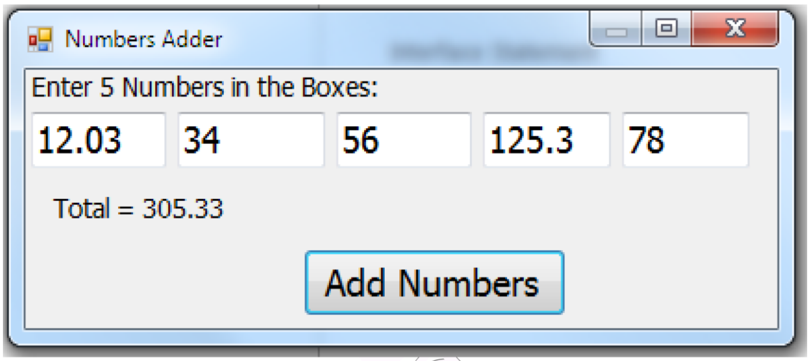
লক্ষ করুন: ByVal ParamArray args() As Double দ্বারা আমরা বুঝাচ্ছি যে, ফাংশনের প্যারামিটার বা আর্গুমেন্ট হবে একটি এ্যারে ভেরিয়েবল। এর টাইপ ও রিটার্ন টাইপ হবে ডাবল। এই ফাংশনের মান ফাংশনের নাম দ্বারাই হবে। এজন্য আমরা প্রথম লাইনে নাম লিখে তার মান ০ বানিয়েছি। দ্বিতীয় লাইনে আমরা নিশ্চিত করছি এ্যারের মধ্যে শূন্য কিংবা এর কম পরিমাণ ইলেমেন্ট থাকলে কিছু না করতে। এরপর ফর-নেস্ট লুপ দ্বারা এ্যারের প্রতিটি ইলেমেন্ট তথা ৫টি সংখ্যার যোগফল বের করে রিটার্ন করছি কলিং লাইনে। আর কলিং হচ্ছে নিচের সাবে। ব্যাস!

```
 'this is the calling sub
 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    On Error GoTo ExitLabel 'error handling/ if user leaves on or more text field empty
    Dim a As Double = TextBox1.Text : Dim b As Double = TextBox2.Text 'put numbers into variables
    Dim c As Double = TextBox3.Text : Dim d As Double = TextBox4.Text 'put numbers into variables
    Dim f As Double = TextBox5.Text 'put numbers into variables
    Dim SumOfNumber = calcSum(a, b, c, d, f) 'function call
    Label2.Text = "" 'empty out the result
    Label2.Text = "Total = " & SumOfNumber 'show result
    ExitLabel: 'error jump
End Sub
```

৮ম পরিচ্ছেদ: ***More on Subs & Functions***

সাব ও ফাংশন : ***Subs & Functions***

এবার প্রোগ্রামটি রান করে দেখা যাক, কেমন? নিচের চিত্রে প্রোগ্রাম চলাকালীন জানালা দেখা যাচ্ছে।



লক্ষ করুন: টেক্সট বাক্সগুলোতে আস্ত এবং ডেসিমেল সংখ্যা ঢুকাতে পারবেন। তবে একটিও বাক্সে কিছু না থাকলে ফলাফল কিছুই পাবেন না।

যাক, আমরা এই সলিউশন দ্বারা দেখালাম, ফাংশনে এ্যারে ভেরিয়েবল আর্গুমেন্ট হিসাবে ব্যবহার সম্ভব। আমাদের ফাংশনের আর্গুমেন্ট বা প্যারামিটার হিসাবে আমরা পাঁচটি সংখ্যা দেখিয়েছি: calcSum(a, b, c, d, f)। বাক্সের সংখ্যাকে এ্যারের আইটেম হিসাবে পরিবর্তন করে ফাংশন কল করে ফলাফল নিয়ে এসেছি। একটা ব্যাপার হয়তো আপনার নিকট পরিষ্কার নয়। তাহলো এই কোড লাইনটি: For i As Integer = 0 To UBound(args, 1)। বাস্তবে ব্যাপারটা তেমন জটিল নয়। কিওয়ার্ড এর অর্থ হলো Upper Bound। অর্থাৎ এ্যারের ব্যাপ্তি হবে ১ থেকে [এক্ষেত্রে] ৫। অন্যকথায় লুপের মধ্যে **i** যখন ৪ হবে তখন UBound(args, 1) এর ফলও হবে ৫। সুতরাং ৫ বার মাত্র ফর-নেক্সট লুপ ঘুরবে এবং সংখ্যাগুলো যোগ করে যাবে। এখানে এই টেকনিক ব্যবহারের মূল কারণ হলো এ্যারোকে প্রয়োজনে ছোট বড় করা যাবে। ফাংশনে কোনো কাজ করতে হবে না।

৯ম পরিচ্ছেদ: *Ready-made functions and methods*

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

এ পরিচ্ছেদে আমরা বানানো কিছু ফাংশন ও মেথড নিয়ে উদাহরণসহ আলোচনা করবো। ভিজুয়্যাল বেসিক.নেট এর সৃষ্টিকারীরা প্রোগ্রামারদের সুবিধার জন্য প্রচুর ফাংশন ও মেথড তৈরি করে সিস্টেমে যুক্ত করে দিয়েছেন। এগুলো বিভিন্ন নাম ও নিয়মে ব্যবহার করা যায়। প্রথমে আমরা এরূপ কিছু ফাংশন উদাহরণসহ নিম্নে উপস্থাপন করছি।

ক. String Functions - স্ট্রিং ফাংশন্স -লেখা সম্পর্কিত ফাংশন

১. Split - মনে করুন: 'বাগানে অনেক আকর্ষণীয় ফুল ফুটেছে।' এ বাক্যটির প্রত্যেক শব্দ আলাদা করবেন। কোডিংয়ের মাধ্যমে এরূপ করতে চাইলে আপনি ব্যবহার করতে পারেন Split ফাংশনটি।

উদাহরণ:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
 TextBox1.Text = "" 'clean text box
  Dim A_String As String = "evMv‡b A‡bK AvKl©Yxq dzj dz‡U‡Q|"
  'split the string based on spaces and put them in an Array variable
  Dim Words As String() = A_String.Split(New Char() {" "})
  'Display words in a text box in seperate lines
  For A = 0 To Words.Length - 1 'Words.Length gives number of words in Array
     TextBox1.Text = TextBox1.Text & Words(A) & vbCrLf
  Next
End Sub
```

লক্ষ করুন: Split ফাংশন দিয়ে লেখার মধ্যে মোট ক'টি শব্দ আছে তা সহজে বের করা যায়। যেমন:

```
Dim Words As String() = A_String.Split(New Char() {" "}
MsgBox(Words.Length) 'this will show number of words
```

২. Join -আলাদা স্ট্রিং ইচ্ছে মতো একত্রিত করা যায় এই ফাংশন দিয়ে। যেমন:

```
'Join function at work
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
   Dim An_Array(4) As String 'declare an array of 5 words
   An_Array(0) = "evMv‡b" : An_Array(1) = "A‡bK" : An_Array(2) = "AvKl©Yxq"
   An_Array(3) = "dzj" : An_Array(4) = "dz‡U‡Q|" 'put words into them
   Dim JoinText As String = String.Join(" ", An_Array) 'using space as delimiter
   TextBox1.Text = JoinText 'display result
End Sub
```

৩. Compare -দু'টি স্টিং এর মধ্যে পার্থক্য নির্ণয় সময় সময় জরুরী হয়ে ওঠে। ভিজুয়্যাল বেসিক এ ফাংশন তৈরি করেছে পার্থক্য নির্ণয়ের জন্য। যেমন:

```
Dim StA As String = "a" : Dim StB As String = "b"
Dim C As Integer = String.Compare(StA, StB) : MsgBox(C) 'gives -1
C = String.CompareOrdinal(StB, StA) : MsgBox(C) 'gives 1
C = StB.CompareTo(StA) : MsgBox(C) 'gives 1
C = "X".CompareTo("X") : MsgBox(C) 'gives 0
```

এখানে: -১ অর্থ প্রথমটি ছোট, ১ অর্থ দ্বিতীয়টি বড় এবং ০ অর্থ উভয়টি সমান। যে কোনো শব্দ বা বাক্যও সমান কি না এই ফাংশন দ্বারা পরীক্ষা করা যায়। পরের পৃষ্ঠায় আরেকটি উদাহরণ দেখুন।

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

```
Dim Word1 As String = "আসমান" : Dim Word2 As String = "প্রথম"
    'comapre
If String.Compare(Word1, Word2, False) = 0 Then
        MsgBox("String Match")
   Else
     MsgBox("String does not Match") 'this message appears
   End If
```

৪. Insert -এই ফাংশনের কাজ হলো স্ট্রিং এর ক্যারাক্টার পজিশনে প্রদত্ত স্ট্রিং ঢুকানো। যেমন:

```
' Insert the word [and].
   Dim names As String = "Romeo Juliet"
   Dim Shakespeare As String = names.Insert(6, "and ") ' 'and ' inserted in the space (character    'position
6: result: "Romeo and Juliet"
```

৫. Length -এই ফাংশনের কাজ হলো স্ট্রিং এর ভেতর ক'টি ক্যারাক্টার আছে তা নির্ণয় করা: যেমন:

```
Dim value As String = "dotnet"
Dim length As Integer = value.Length 'gives length = 6
value += ".aziz"  'now value is 'dotnet.aziz'
length = value.Length 'gives length = 11
```

৬. Remove -এটা দ্বারা কোনো স্ট্রিং এর প্রদত্ত পজিশন থেকে বাকি সবগুলো মুছে ফেলা যায়। যেমন:

```
' Completely incorrect input string.
  Dim value As String = "Engineer Azizul Bari wrote 'Visual Basic.Net Programming'. and.."
  'We need to remove ' and..' part, eg. everything after last SPACE
  Dim index As Integer = value.LastIndexOf(" "c)
  ' Remove everything starting at that position.
  value = value.Remove(index)
  TextBox1.Text = value
```

৭. Replace -এ ফাংশনের কাজ হলো যে কোনো স্ট্রিং - যেমন কোনো ডকুমেন্টের টেক্সটে একই শব্দ বা লাইন ইত্যাদি যতোবারই থাকুন না কেনো, তা এক সঙ্গে বদলানো। যেমন:

```
Dim FstString As String = "আমরা এই লেখার তাই শব্দ বদলাতে চাই। তাই, তাই"
     ঋংঃঋঃৎরহম = ঋংঃঋঃৎরহম.জবঢ়ষধপব("তাই", "")
    MsgBox(FstString) 'gives 'আমরা এই লেখার শব্দ বদলাতে চাই।'
```

৮. Substring -এটা একটি সচরাচর ব্যবহৃত ফাংশন। এর কাজ হলো যে কোনো লেখা থেকে প্রদত্ত অক্ষরসমূহ সরিয়ে নিয়ে প্রদর্শন করা। যেমন:

```
'Use this string literal for the demonstration.
 Dim literal As String = "Our Our country is Bangladesh." -আমরা 'Our' শব্দটি বের করবো।
 Dim substring As String = literal.Substring(0, 4) 'first word
MsgBox(substring) 'will show - Our country is Bangladesh.
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

৯. Contains -এটা দ্বারা জানা যায় একটি স্টিং এর মধ্যে আরেকটি স্ট্রিং আছে কি না। ফলাফল True বা False। যেমন:

```
Dim literal As String = "লেখা থেকে প্রথম শব্দটি বের করবো।" : Dim newString As Boolean = literal.Contains("প্রথম")
    MsgBox("প্রথম: " & newString) 'result True : Dim newStringt As Boolean = literal.Contains("আসমান") : MsgBox("আসমান: " & newStringt) 'result False
```

১০. InStr -এই ফাংশন দিয়ে একটি স্ট্রিং এর ভেতর আরো একটি স্ট্রিং আছে কি না পরীক্ষা করা যায়। যেমন:

```
Dim literal As String = "লেখা থেকে প্রথম শব্দটি বের করবো।"
Dim isString As Integer = InStr(literal, "বের", CompareMethod.Text)
MsgBox(isString) 'returns 24 -found at character position 24
isString = InStr(literal, "আমার", CompareMethod.Binary)
MsgBox(isString) 'retuns 0 -string not found
```

১১. Mid -এটি দিয়ে স্ট্রিং এর ভেতর থেকে কাঙ্ক্ষিত স্ট্রিং বের করা যায়। যেমন:

```
Dim literal As String = "লেখা থেকে প্রথম শব্দটি বের করবো।"
Dim NString As String = Mid(literal, 6, Len(literal))
MsgBox(NString) 'result-  থেকে প্রথম শব্দটি বের করবো।
```

লক্ষ করুন: Length ফাংশন ও উপরে ব্যবহৃত Len ফাংশন এর কাজ একই। প্রথমটি ভিজুয়্যাল বেসিক-কে ভিজুয়্যাল বেসিক.নেট -এ উন্নীতকরণকালে সংযুক্ত করা হয়েছে। তবে উভয়টিই কার্যকরী।

১২. একটিমাত্র ক্যারাক্টার বারবার লেখা: নিচের কোডটি এ কাজ করে।

```
Dim txt As New String("স"c, 5)
MsgBox(txt) -ফলাফল -সসসসস
```

১৩. ToUpper & ToLower -এ ফাংশন দ্বারা দু'টো শুধুমাত্র ইংরেজি অক্ষরসমূহ বড় এবং ছোট করা যায়। যেমন:

```
Dim word As String = "ViSuAL bASiC"
MsgBox(word.ToUpper) 'gives VISUAL BASIC
MsgBox(word.ToLower) 'gives visual basic
```

নোট: বাংলা ইউনিকোড লেখা হলে উভয় ফাংশন কাজ করে না। লেখা অপরিবর্তিত থাকবে।

১৪. স্ট্রিং -এর সকল অক্ষর আলাদা করে বের করা: নিচের কোডটুকু এ কাজ করবে। লক্ষণীয়- স্পেইস বা শূন্যস্থানকেও কোড ক্যারাক্টার হিসাবে সনাক্ত করে।

```
Dim AworD As String = "I am a programmer."
For i As Integer = 0 To AworD.Length - 1 'Or 0 to len(AworD) -1
    MsgBox(AworD.Chars(i)) 'shows each character of the sentence
Next
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

১৫. IsDigit -সময় সময় আমাদের জানতে হয় ক্যারাক্টার সংখ্যা [০-৯] কি না। এ ফাংশন দ্বারা তা জানা যায়। যেমন:

```
Dim EntTx As String = TextBox1.Text
   If EntTx <> "" Then
        MsgBox(Char.IsDigit(EntTx.Chars(2)))
   'যদি তৃতীয় অক্ষর সংখ্যা হয়- ফলাফল True। অন্যথায় False হবে।
 End If
```

অনুরূপ আরো দু'টি ফাংশন আছে: (ক) IsNumeric -এটা পুরো লেখাটি সংখ্যা (ডেসিমেলসহ) কি না দেখবে। (খ) IsNumer(Char, Character Position) -এটা ঠিক IsNumeric-এর মতো, তবে চিহ্নিত একটিমাত্র ক্যারাক্টার ইউনিকোড হলেও সনাক্ত করবে। যেমন বাংলা ইউনিকোড ক্যারাক্টার ৯ এ ফাংশন সংখ্যা হিসাবেই সনাক্ত করবে। দৃষ্টান্ত:

```
Dim AString As String = "345.03"
If IsNumeric(AString) = True Then MsgBox("Yes") 'result Yes because whole string is a double
If Char.IsNumber(AString, 3) = True Then
   MsgBox("Yes")
Else
   MsgBox("No") 'result No because . isn't a number.
End If
AString = "৩৯৫.০৩"□বাংলা ইউনিকোড সংখ্যা□
□□□□□□□□If IsNumeric(AString) = True Then
   MsgBox("Yes")
Else
   MsgBox("No") 'result No because IsNumeric doesn't recognize unicode
End If
If Char.IsNumber(AString, 1) = True Then
   MsgBox("Yes") 'result Yes because index 1 = 9 [Remember: index always 0 based]
Else
   MsgBox("No")
End If
```

১৬. Trim, LTrim, RTrim -এ তিনটি ফাংশন যথাক্রমে লেখার উভয় দিকের, বায়ের ও ডানের শূন্যস্থান [স্পেইস] মুছে ফেলে। যেমন:

```
Dim AString As String = " 345.03 is a number. "
   AString = (Trim(AString)) 'both sides space erased
   AString = (LTrim(AString)) 'left side space erased
   AString = (RTrim(AString)) 'right side space erased
   TextBox1.Text = AString 'Clean text: '345.03 is a number.'
```

খ. Conversion Functions - কনভারশন ফাংশন্স -রূপান্তর সম্পর্কিত ফাংশন

১. Asc -এই ফাংশন দিয়ে যে কোনো ক্যারাক্টারের সংখ্যা কোড জানা যায়। যেমন:

```
Dim codeInt As Integer
' The following line of code sets codeInt to 65.
codeInt = Asc("A")
' The following line of code sets codeInt to 97.
codeInt = Asc("a")
' The following line of code sets codeInt to 65.
codeInt = Asc("Apple")
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

২. AscW -এই ফাংশন দিয়ে যে কোনো ইউনিকোড ক্যারাক্টারের সংখ্যা কোড জানা যায়। কোড সংখ্যার ব্যাপ্তি হলো ০ থেকে ৬৫৫৩৫। উদাহরণ:

```
AscW("প") 'returns 2474
AscW("ক") 'returns 2453
```

৩. Chr -এ ফাংশন সংখ্যা কোড থেকে ক্যারাক্টার বের করে। যেমন:

```
Chr -retuns character
Dim associatedChar As Char
' Returns "A".
associatedChar = Chr(65)
' Returns "a".
associatedChar = Chr(97)
' Returns ">".
associatedChar = Chr(62)
' Returns "%".
associatedChar = Chr(37)
```

৪. ChrW -এ ফাংশন সংখ্যা কোড থেকে ইউনিকোড ক্যারাক্টার বের করে। যেমন:

```
ChrW(2474) returns 'প'
ChrW(2453) returns 'ক'
```

৫. Format -এটি সাধারণত তারিখ ফরম্যাট করার জন্য ব্যবহৃত হয়। যেমন:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim TestDateTime As Date = #1/27/2001 5:04:23 PM#
    Dim TestStr As String
    TestStr = Format(Now(), "Long Time") ' Returns current system time in the system-defined long time format.
    TextBox1.Text += TestStr & vbCrLf ' Also returns current system date in the system-defined long date format, using the single letter code for the format.
    TestStr = Format(Now(), "D")
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(TestDateTime, "h:m:s") ' Returns the value of TestDateTime in user-defined date/time formats. Returns "5:4:23".
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(TestDateTime, "hh:mm:ss tt") ' Returns "05:04:23 PM".
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(TestDateTime, "dddd, MMM d yyyy") ' Returns "Saturday, Jan 27 2001".
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(TestDateTime, "HH:mm:ss") ' Returns "17:04:23".
    TextBox1.Text += TestStr & vbCrLf
    TestStr = Format(23) ' Returns "23"
    TextBox1.Text += TestStr
End Sub
```

বায়ের প্রোগ্রাম রান করার ফলাফল:

```
12:50:54 PM
Wednesday, August 10, 2016
5:4:23
05:04:23 PM
Saturday, Jan 27 2001
17:04:23
23
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

অবশ্য ফাংশন দ্বারা ব্যবহারকারীর ইচ্ছেমাফিক কাজও করা যায়। যেমন নিচের কোড দ্বারা সংখ্যা ফরম্যাট করা যাবে:

```
Dim TestStr as String
TestStr = Format(54590125.4, "###,##,##0.00") 'Results 54,590,125.40
```

গ. Math Functions - ম্যাথ ফাংশন্স - গাণিতিক ফাংশন্স। এসব ফাংশনের নামের পূর্বে Math. শব্দ লিখতে হবে।

১. Math.Abs -এ ফাংশন যে কোনো সংখ্যাকে পজিটিভ সংখ্যায় পরিণত করে। যেমন:
Math.Abs(-123) -returns 123

২. Math.Acos -এটা কোণের Cosine সংখ্যা থেকে ডিগ্রী রেডিয়ান দেবে। যেমন:
Math.Acos(0.9568) -returns 0.295 radians

৩. Math.Asin -এটা কোণের Sine সংখ্যা থেকে ডিগ্রী রেডিয়ান দেবে। যেমন:
Math.Asin(0.145268) -returns 0.14578 radians

অন্যান্য হলো: Atan [Arc Tan], Cos [Cosine], Cosh [Hyperbolic Cosine], Sin [Sine], Sinh [Hyperbolic Sine], Tan [Tangent], Tanh [Hyperbolic Tan], Exp [Exponential -e^y to the power of y], Pow [x^y Value], Log [Logarithm -base 2], Log10 [Logarithm -base 10], Sqrt [Square Root], Sign [Integer value with + or - Sign], Min [Smaller of two numbers], Max [Larger of two numbers]

আরো কিছু ম্যাথ ফাংশন হলো:

(ক) `Math.BigMul(1256, 2154) -returns 2705424`

(খ) `Math.IEEERemainder(45,7) -returns (45/7) reminder 3`

(গ) `Math.Round(45.8) -returns 46; Math.Round(45.81255, 3) -returns 45.823`

(ঘ) `Math.Truncate(45.81255) -returns 45`

ঘ. Type Conversion Functions -টাইপ (ভেরিয়েবল ধরন) রূপান্তর ফাংশন

১. Integer > < Boolean - উদাহরণ:

```
Dim a, b, c As Integer
Dim check As Boolean
a = 5 : b = 5 ' The following line of code sets check to True.
check = CBool(a = b) 'check is True
c = 0 ' The following line of code sets check to False.
check = CBool(c) 'check is False
```

২. Double > < Byte - উদাহরণ:

```
Dim aDouble As Double : Dim aByte As Byte
aDouble = 125.5678 ' The following line of code sets aByte to 126.
aByte = CByte(aDouble)
```

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

৩. String > < Char, CChar - উদাহরণ:

```
Dim aString As String
Dim aChar As Char
aString = "BCD" ' CChar converts only the first character of the string.
aChar = CChar(aString) 'sets aChar to "B".
```

৪. String > < Integer, Char - উদাহরণ:

```
Dim someDigits As String
Dim codePoint As Integer
Dim thisChar As Char
someDigits = InputBox("Enter code point of character:") 'ask the user
codePoint = CInt(someDigits) 'convert it to integer and equate ti declared integer type
' The following line of code sets thisChar to the Char value of codePoint.
thisChar = ChrW(codePoint)
```

৫. String > < To Date and Time -উদাহরণ:

```
Dim aDateString, aTimeString As String
Dim aDate, aTime As Date
aDateString = "February 12, 1969"
aTimeString = "4:35:47 PM" ' The following line of code sets aDate to a Date value.
aDate = CDate(aDateString) ' The following line of code sets aTime to Date value.
aTime = CDate(aTimeString)
```

৬. Double > < To Integer -উদাহরণ:

```
Dim aDbl As Double
Dim anInt As Integer
aDbl = 2345.5678 ' The following line of code sets anInt to 2346.
anInt = CInt(aDbl)
```

৭. Double > < To Object -ভিজুয়্যাল বেসিক.নেটে Object -একটি টাইপ ভেরিয়েবল। উদাহরণ:

```
Dim aDouble As Double
Dim anObject As Object
aDouble = 2.7182818284 ' The following line of code sets anObject to a pointer to
aDouble.
anObject = CObj(aDouble)
```

দ্র: উপরের কোড অংশের কমেন্টে pointer -শব্দটি ব্যবহার করেছি। pointer কি? প্রোগ্রামিং ভাষায় pointer হলো: প্রোগ্রামিং ভাষার বস্তু [Object], যার মান [Value] ইঙ্গিতে নির্দেশ করে [Points to] কম্পিউটারের অন্যত্র মেমোরি ঠিকানায় সংরক্ষিত মানের দিকে। আরো বুঝতে আমরা গ্রন্থের 'সূচিপত্রের' কথা ভাবতে পারি। সূচিতে সংশ্লিষ্ট বিষয়ের উপর আলোচনার পৃষ্ঠা সংখ্যা থাকে। অন্যকথায়, বইয়ের সূচিতে দেওয়া এ পৃষ্ঠা নাম্বার হলো একটি পয়েন্টার যা আমাদেরকে জানিয়ে দেয় কোন্ পৃষ্ঠায় তথ্যাদি সংরক্ষিত [লিখিত] হয়েছে। পয়েন্টার ব্যবহারের মাধ্যমে কম্পিউটিং গতি বৃদ্ধি পায়। কারণ, একই টিকানায় সংরক্ষিত তথ্য বিভিন্ন কোড লাইনে ব্যবহৃত হয়ে থাকে।

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

৮. String > < Char, CChar - উদাহরণ:

Dim aString As String
Dim aChar As Char
aString = "BCD" ' CChar converts only the first character of the string.
aChar = CChar(aString) 'sets aChar to "B".

৮. Double > < Byte, Short - উদাহরণ:

Dim aDouble As Double
Dim anSByte As SByte
aDouble = 39.501 ' The following line of code sets anSByte to 40.
anSByte = CSByte(aDouble)
Dim aByte As Byte
Dim aShort As Short
aByte = 100 ' The following line of code sets aShort to 100.
aShort = CShort(aByte)

Byte data type -কম্পিউটারের মেমোরিতে সংরক্ষিত সর্বাধিক স্বল্পমাত্রার সাংখ্যিক তথ্যকে বলে বিট [Bit]। একেক বিটকে বিজ্ঞানীরা দু'টি মাত্র সাংখ্যিক কোড দ্বারা উপস্থাপন করেন। এগুলো হলো: ০ এবং ১। এ দু'টোর সমন্বয়ে ঘটিত ৮টি বিট হলো একেকটি বাইট [Byte]। সুতরাং ১ বাইট মানে ৮ বিট তথ্য।

৯. Double > < String - উদাহরণ:

Dim aDouble As Double
Dim aString As String
aDouble = 437.324 ' The following line of code sets aString to "437.324".
aString = CStr(aDouble)

১০. Double > < String - উদাহরণ:

Dim testNumber As Long = 1000 ' The following line of code sets testNewType to 1000.0.
Dim testNewType As Single = CType(testNumber, Single)

৬. Properties -ভিজুয়্যাল বেসিকে প্রোপার্টি

১. DateString -কম্পিউটারের সেটিং মুতাবিক বর্তমান তারিখ ও সময়। ব্যবহার:

MsgBox("The current date is " & DateString) 'returns 08-11-2016
MsgBox("The formatted date is " & Format(Now, "dddd, d MMM yyyy")) 'returns Thursday, 11 August 2016.

২. Now -এখনকার তারিখ ও সময়। ব্যবহার:

Dim ThisMoment As Date ' The following statement calls the Get procedure of the Visual Basic Now property.
ThisMoment = Now : MsgBox(ThisMoment) 'retuns 8/11/2016 12:44:03 PM

৯ম পরিচ্ছেদ: ***Ready-made functions and methods***

বানানো ফাংশন ও মেথড: *Ready-made functions & Methods*

৩. ScriptEngine -বর্তমান রান-টাইমের ওপর তথ্য। উদাহরণ:
MsgBox(ScriptEngine) 'retuns- VB

৪. ScriptEngineBuildVersion -বর্তমান রান-টাইমের বিল্ড ভার্শনের ওপর তথ্য। উদাহরণ:
MsgBox(ScriptEngineBuildVersion) 'retuns- 50709

৫. ScriptEngineMajorVersion -বর্তমান রান-টাইমের মেজর ভার্শনের ওপর তথ্য। উদাহরণ:
MsgBox(ScriptEngineMajorVersion) 'returns 11

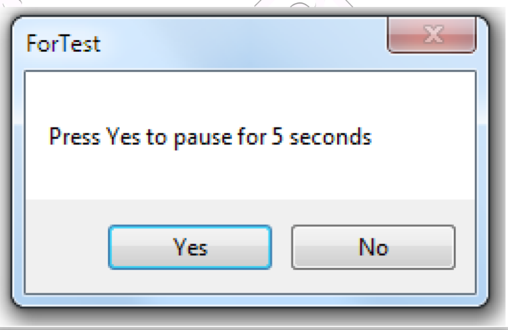
৬. ScriptEngineMinorVersion -বর্তমান রান-টাইমের মাইনোর ভার্শনের ওপর তথ্য। উদাহরণ:
MsgBox(ScriptEngineMinorVersion) 'returns 0

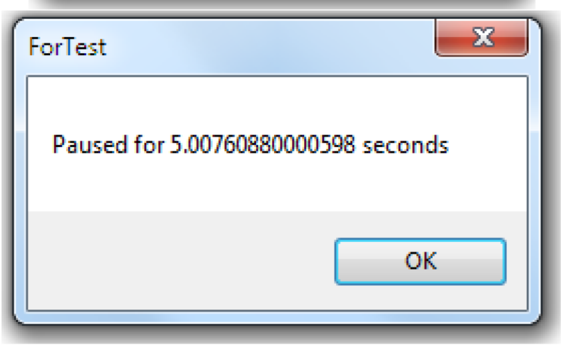
৭. TimeOfDay -দিনের বর্তমান সময়ের ওপর তারিখ ভ্যালু সেট বা রিটার্ন করে। উদাহরণ:

```
Dim currentTime As Date
currentTime = TimeOfDay 'returns 1:10:19 PM
```

৭. Timer -আগের মধ্যরাত থেকে বর্তমান পর্যন্ত কতো সেকেন্ড অতিবাহিত হয়েছে তার একটি মাত্রা। উদাহরণ:

```
Public Sub waitFiveSeconds()
    If TimeOfDay >= #11:59:55 PM# Then
        MsgBox("The current time is within 5 seconds of midnight" &
            vbCrLf & "The timer returns to 0.0 at midnight")
        Return
    End If
    Dim start, finish, totalTime As Double
    If (MsgBox("Press Yes to pause for 5 seconds", MsgBoxStyle.YesNo)) = MsgBoxResult.Yes Then
        start = Microsoft.VisualBasic.DateAndTime.Timer
        ' Set end time for 5-second duration.
        finish = start + 5.0
        Do While Microsoft.VisualBasic.DateAndTime.Timer < finish
            ' Do other processing while waiting for 5 seconds to elapse.
        Loop
        totalTime = Microsoft.VisualBasic.DateAndTime.Timer - start
        MsgBox("Paused for " & totalTime & " seconds")
    End If
End Sub
```





উপরের প্রোগ্রাম রানিংয়ের চিত্র।

১০ম পরিচ্ছেদ: *Object Oriented Programming [OOP]* -বস্তুভিত্তিক প্রোগ্রামিং

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

শিক্ষার্থীরা অবজেক্ট অরিয়েন্টেড প্রোগ্রামিং [OOP] কথাটি ইতোমধ্যে অন্যত্র দেখেছেন। এ সম্পর্কে বিস্তারিত আলোচনা জরুরী মনে করে এ পরিচ্ছেদ লেখার প্রয়াস। কম্পিউটার সফ্টওয়্যার সৃষ্টির জগতে OOP একটি আধুনিক ধারণা [concept]। ভিজুয়্যাল বেসিক.নেট এর পূর্বসুরী ছিলো শুধু ভিজুয়্যাল বেসিক। ভার্শন ৬.০ এর পর মাইক্রসফ্ট ভিজুয়্যাল বেসিক আর ডেভোলাপ করে নি। ভিজুয়্যাল বেসিক.নেট এর ওপরই মনোযোগ দিয়েছে। এর একটি অন্যতম কারণ, আধুনিক অন্যান্য ভাষা [যেমন c++, Java] হলো পূর্ণাঙ্গ OOP-নিয়ন্ত্রিত। মাইক্রসফ্ট ভিজুয়্যাল বেসিকের মতো নতুন একটি ভাষা তৈরি করতে সচেষ্ট হয় পৃথিবীব্যাপী অসংখ্য ভক্ত-অনুরক্ত প্রোগ্রামারদের সুবিধার্থে। সবাই চাচ্ছিলো একটি উন্নতমানের ভাষা যাতে সরলতা থাকবে- যেরূপ মূল ভিজুয়্যাল বেসিকে ছিলো, এসাথে আধুনিক অন্যান্য ভাষার সঙ্গে সামঞ্জস্যশীলতা নিশ্চিত করা। আর এরই ফলাফল স্বরূপ আমরা পেয়েছি পূর্ণাঙ্গ OOP-সহ মাইক্রসফটের এই .নেট ভাষাটি। এই ভাষার আরেক বৈশিষ্ট্য হলো উইন্ডোজ অপারেটিং সিস্টেমের ফাংশন, মেথড ইত্যাদি সরাসরি কাজে লাগানোর সুযোগ। এ সুযোগের নামই হলো API- Application Program Interface। মূলত একটি কোডিং সিস্টেম যেখানে সফটওয়্যার তৈরিকারী নিজের কোড দ্বারা সিস্টেমের কোডের সঙ্গে যোগাযোগ সৃষ্টি করতে পারেন।
যাক, ফিরে আসি OOP প্রোগামিং কনসেপ্টে। আমাদের এ জগতটি মূলত বস্তু বা অবজেক্টের তৈরি। একটি বাড়ি। এটা সার্বিকভাবে একটি বস্তু যা সৃষ্টি হয়েছে একাধিক বিভিন্ন অংশের সমন্বয়ে। যেমন: ইট, পাথর, বালি, দেওয়াল, ফ্লোর, কক্ষ, ছাদ, পিলার ইত্যাদি। এ-সবই বাড়ির অংশ বা প্রোপার্টি। অনুরূপ একটি বৃক্ষের মধ্যে বেশ ক'টি প্রোপার্টি চিহ্নিত করা যায়। যেমন: এর মূল, কাণ্ড, শাখা, প্রশাখা, পত্র, উচ্চতা, কাণ্ডের ব্যাস ইত্যাদি। কম্পিউটার বিজ্ঞানীরা এরূপ প্রাত্যহিক বস্তু ও এদের প্রোপার্টি গবেষণা করে OOP সৃষ্টি করেছেন প্রোগ্রামিং সুবিধার্থে।

যে কোনো প্রোগ্রামিং ভাষাকে OOP হিসেবে স্বীকৃতি পেতে হলে তার মধ্যে তিনটি মৌলিক জিনিস থাকতে হবে:

ক. **Encapsulation** -ঠিক যেরূপ ক্যাপসুলের ভেতর কিছু থাকে সেরূপ কোডগুলো বন্ধ অবস্থায় থাকা। অন্যকথায় বাইরের জগত থেকে গোপন রাখা।

খ. **Inheritance** -ঠিক যেরূপ কোনো প্রাণী নিজের বংশবিস্তার লাভে সমর্থ, সেরূপ নিজেকে কপি করার ক্ষমতা থাকা। অর্থাৎ উত্তরাধিকার ব্যবস্থা থাকতে হবে।

গ. **Polymorphism** -ঠিক একই দলভুক্ত বিভিন্ন বস্তুর মতো থাকতে হবে, একই দলভুক্ত একাধিক ভিন্ন বস্তু।

এই তিনটি বিষয়ের ওপর আরো কিছু ব্যাখ্যা তুলে ধরার প্রয়াস পাচ্ছি। এগুলো বুঝে এসে গেলে OOP সম্পর্কে ধারণাও আরো স্পষ্ট হয়ে ওঠবে।

ক. Encapsulation

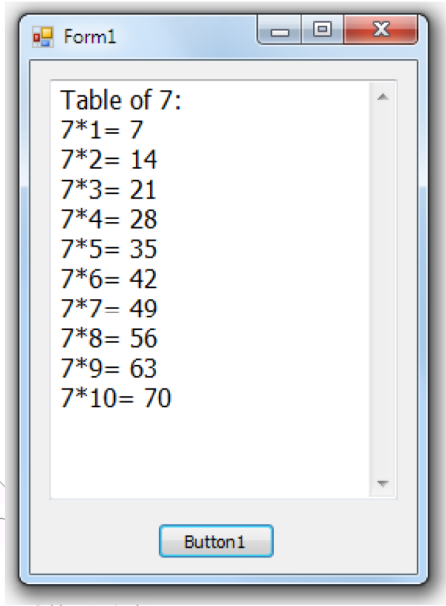
কোডিংয়ের ক্ষেত্রে বার বার ব্যবহারযোগ্য এবং ভিন্ন এ্যাপলিকেশনে বহনযোগ্য কিছু ইউনিট তৈরি করা যায়। এগুলোকে বলে আত্মনির্ভরশীল মডিউল [self-contained module]। এসব মডিউলে থাকে তথ্য প্রসেস করার বিভিন্ন ফাংশন। ব্যবহারকারী কর্তৃক নিরূপণকৃত তথ্য-ধরনকে [user-defined data type] বলে ক্লাস [class]। প্রত্যেক ক্লাসে থাকে তথ্য [data]। আরো থাকে এক বা একাধিক পদ্ধতি [method] যাদের মাধ্যমে তথ্যাদি রদবদল করা যায়। ক্যাপসুলের মতো আত্মনির্ভরশীল আবদ্ধ ক্লাসের ডাটা অংশকে বলে সময়ের পরিবর্তনশীল তথ্য বা ইন্সট্যান্ট ভেরিয়েবল্স [instant variables]। আর ক্লাসের একটি ইন্সস্ট্যান্ট হলো একটি বস্তু [object]। আমরা এগুলোকে পাবলিক লাইব্রেরি সিস্টেমের সঙ্গে তুলনা করতে পারি। ক্লাস হতে পারে লাইব্রেরির সদস্য এবং আমির ও সাকিয়া হতে পারে এ লাইব্রেরি ক্লাসের দু'টি ইন্সট্যান্ট [দু'টি বস্তু]। আশারাখি এ কথাগুলো পাঠ করে আপনি Encapsulation সম্পর্কে আরো কিছুটা স্পষ্ট ধারণা পেয়ে গেছেন। তবে বাস্তব ক্ষেত্রে একটি কোড উদাহরণ দেখুন। এতে বিষয়টি আরো স্পষ্ট হবে।

১০ম পরিচ্ছেদ: *Object Oriented Programming [OOP] -বস্তুভিত্তিক প্রোগ্রামিং*

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

ভিজুয়্যাল বেসিক.নেটে **Encapsulation** ব্যবহারের উদাহরণ: একটি গুণের নামতা:

```
Public Class Form1
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    TextBox1.Text = "" 'clear text
    MySub() 'calling another sub
   End Sub
    'called from button clicked event sub
   Public Sub MySub()
      Dim obj As ClassA = New ClassA(7) 'create object of ClassA
      obj.table() 'calling member method by object [function call which is in classA]]
   End Sub
End Class
====
'class declaration
Public Class ClassA 'ClassA starts
   Private Value As Integer 'private member variable
   Sub New(V As Integer) 'constructor with 1 argument
      Value = V
   End Sub
   Public Function table() 'member method to show member variable
      Dim Heading, TableV As String
      Heading = "Table of " & Value & ": " & vbCrLf 'heading text String with
return vbCrLf Character
      Form1.TextBox1.Text += Heading 'display heading in the TextBox
      For i As Integer = 1 To 10 Step 1 'calculate times table and display
        TableV = Value & "*" & i & "= " & Value * i & vbCrLf 'calculate and
put into variable
        Form1.TextBox1.Text += TableV 'display '+=' means add text to
existing text.
      Next
      Return Nothing
   End Function
End Class 'classA ends
```



বায়ের প্রোগ্রামের ফলাফল।

খ. Inheritance

ক্লাসগুলো সৃষ্টি হয় হাইআর্কির [hierarchies] সূত্র ধরে। অর্থাৎ ক্রমাধিকারের মাধ্যমে এগুলো অস্তিত্বশীল হয়। ইনহেরিট্যান্স [Inheritance] বা উত্তরাধিকার থাকায় কোড কাঠামো ও মেথড একটি ক্লাস থেকে হাইয়ার্কির নিচ দিকে যেতে সক্ষম হয়। এর ফলে জটিল সিস্টেমেও কম প্রোগ্রামিংয়ের জরুরত হবে। অন্যকথায় ভিন্ন কোনো পদক্ষেপ হাইয়ার্কির নিম্নতম স্তরে যুক্ত করতে হলে পুরো হাইয়ার্কিকে অপরিবর্তিত রেখেই শুধুমাত্র ঐ পদক্ষেপটি তৈরি করে যুক্ত করা সম্ভব হবে। এই পদক্ষেপের বাকি সবকিছুই উত্তরাধিকারসূত্রে পাওয়া যাবে [inherited]। এভাবে প্রস্তুতকৃত বস্তু পুনঃব্যবহারের সুযোগ থাকাটাকে অবজেক্ট প্রযুক্তির ক্ষেত্রে সর্বাধিক বড় সুবিধা হিসাবে ধারণা করা হয়।

একটি ক্লাস যখন অপর আরেকটি ক্লাস Inherit করে বা উত্তরাধিকারসূত্রে প্রাপ্ত হয় তখন একে বলে derived [উদ্ভূত] ক্লাস। অপরদিকে উদ্ভূত ক্লাস থেকে যে ক্লাস উত্তরাধিরসূত্রে প্রাপ্ত হয় তাকে বলে Base Class [মূল ক্লাস]। কেউ কেউ অন্যান্য নামও ব্যবহার করেন, যেমন: প্যারেন্ট [অভিভাবক], চাইল্ড [সন্তান] ইত্যাদি। এবার একটি উদাহরণ দেখুন।

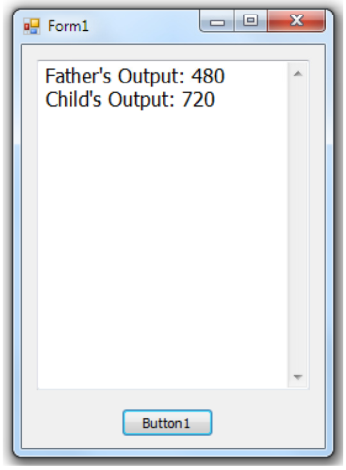
```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        TextBox1.Text = "" 'clear text
        InVokes()
    End Sub
```

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

```
    Sub InVokes() 'method- called from button's click event sub
        Dim obj As Child = New Child(5, 48)
        TextBox1.Text += "Father's Output: " & obj.FatherOutput & vbCrLf
        TextBox1.Text += "Child's Output: " & obj.ChildOutput
    End Sub
End Class
'=======Base Class=======
Public Class Father 'start of base class Father
  Protected Resources As Integer
  Protected WorkInHours As Integer
  Public Sub New(r As Integer, w As Integer) 'constructor with 2 arguments
    Resources = r
    WorkInHours = w
  End Sub

  Public Function FatherOutput() As Integer 'method
    Dim Output As Integer = Resources * WorkInHours
    Return Output
  End Function
End Class
'========Inherited Class=======
Public Class Child : Inherits Father 'derived class
  Private Resources As Integer
  Private WorkInHours As Integer
  Public Sub New(r As Integer, w As Integer) 'constructor with 2 arguments
    MyBase.New(10, 48)
    Me.Resources = r
    Me.WorkInHours = w
  End Sub

  Public Function ChildOutput() As Integer 'method
    Dim Output As Integer = (Me.Resources + MyBase.Resources) * WorkInHours
    Return Output
  End Function
End Class
```



বায়ের প্রোগ্রাম রানের ফলাফল

খ. Polymorphism

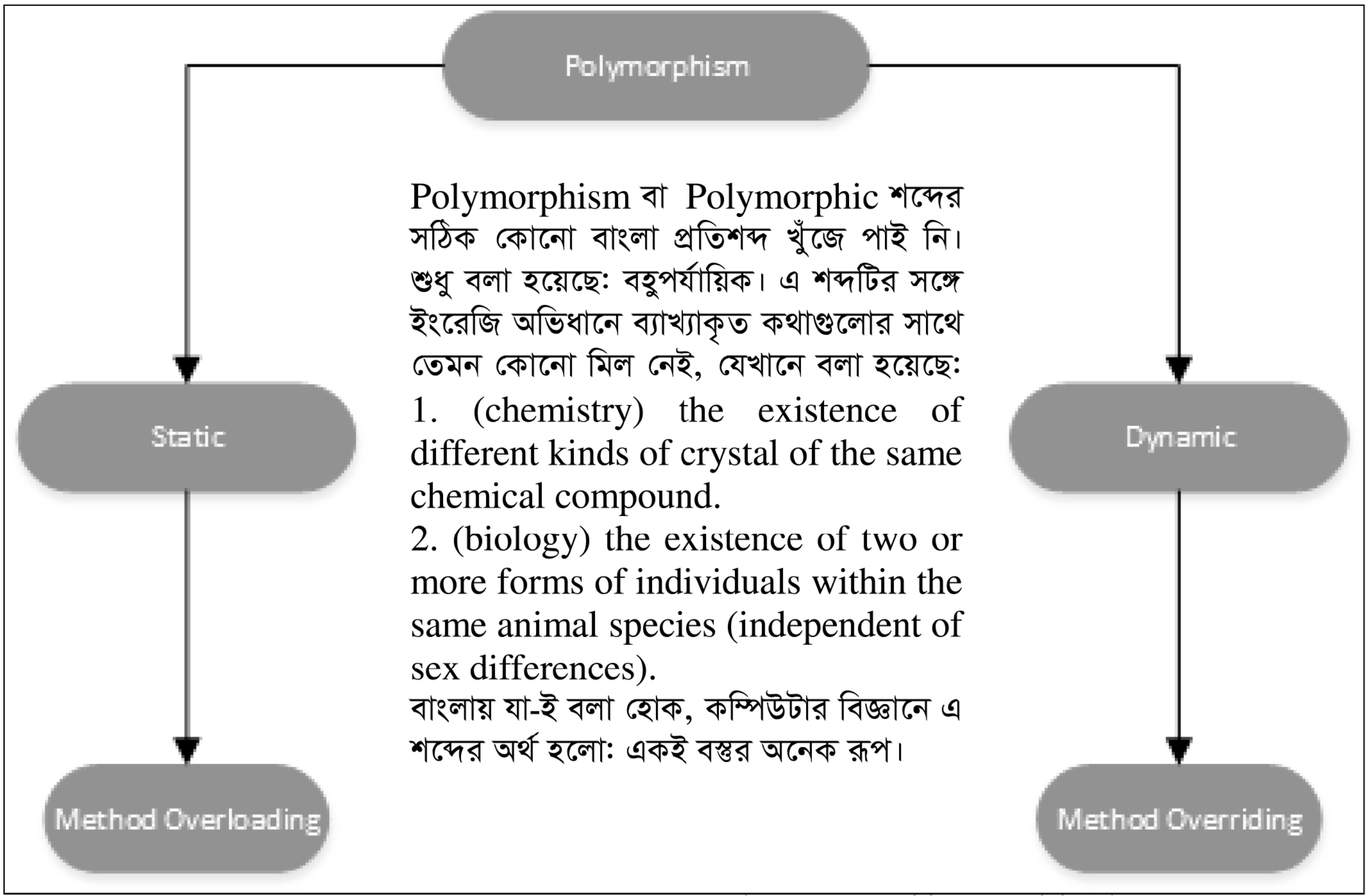
শব্দটি আসলে রসায়ন ও জীববিদ্যা থেকে ধার-করা। রসায়ন শাস্ত্রে Polymorphism-এর অর্থ হলো একই রসায়নভুক্ত কম্পাউন্ড [যুক্ত পদার্থ] দ্বারা সৃষ্ট বিভিন্ন ক্রিস্ট্যাল [স্ফটিক]। জীববিদ্যায় এ শব্দের অর্থ হলো একই প্রজাতিভুক্ত বিভিন্ন প্রাণীর অস্তিত্ব যা মূলত লিঙ্গ নির্ভরশীলহীন।

অবজেক্ট অরিয়েন্টেড প্রোগ্রামিংয়ে বিমূর্ত [abstract] কোড কাঠামো ও মেথড [পদ্ধতি] তৈরি সম্ভব যাদের যথার্থ ধরন বা টাইপ রানটাইম [প্রোগ্রাম চলাকালীন অবস্থা] এর পূর্বে জানা যায় না। যেমন ধরুন, স্ক্রীনের কারসার মাউস চলাকালে তার আকার তীর [এ্যারো] থেকে বদলে রেখায় [লাইন] পরিণত হতে পারে। এটা নির্ভর করবে প্রোগ্রামের ভাবধারার ওপর। সুতরাং আপনি প্রোগ্রামার হিসেবে স্ক্রীনে মাউস চালানোর জন্য লিখবেন 'cursor' রুটিন [কোড বল্ক]। আর পলিমোরফিজম রানটাইমের সময় মাউসের আকার প্রয়োজনমাফিক রদবদল করবে। এটা নতুন এই আকারকে সহজে সম্পৃক্ত করতেও সুযোগ সৃষ্টি করে দেবে। অন্যকথায় রসায়ন শাস্ত্রে যেরূপ একই রসায়নভুক্ত কম্পাউন্ড থেকে সৃষ্টি হতে পারে বিভিন্ন ক্রিস্ট্যাল এবং জীববিদ্যায় একই প্রজাতিভুক্ত বিভিন্ন প্রাণীর অস্তিত্ব সম্ভব- তদ্রূপ অবজেক্ট অরিয়েন্টেড প্রোগ্রামিংয়েও রান-টাইমের সময় একটি অবজেক্ট প্রয়োজনমাফিক তার আকার, আয়তন, কর্ম ইত্যাদিও রদবদল করার ক্ষমতা রাখতে পারে। আর একেই বলে **Polymorphism**। কোডে পলিমোরফিজম্ হয় মেথড অভাররাইডিং ও মেথড অভারলোডিং এর মাধ্যমে। উদাহরণ দ্বারা বুঝা অনেকটা সহজ। [পরের পৃষ্ঠায় দেখুন]

১০ম পরিচ্ছেদ: *Object Oriented Programming [OOP] -বস্তুভিত্তিক প্রোগ্রামিং*

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

Polymorphism -এর দু'টি রূপ আছে: ১. Static Polymorphism [স্থিত পলিমোরফিজম্] ও ২. Dynamic Polymorphism [বেগবান পলিমোরফিজম্]। নিচের ব্লক চিত্র থেকে উভয়টির স্বরূপ সম্পর্কে অতিরিক্ত ধারণা পাওয়া যেতে পারে।



Polymorphism বা Polymorphic শব্দের সঠিক কোনো বাংলা প্রতিশব্দ খুঁজে পাই নি। শুধু বলা হয়েছে: বহুপর্যায়িক। এ শব্দটির সঙ্গে ইংরেজি অভিধানে ব্যাখ্যাকৃত কথাগুলোর সাথে তেমন কোনো মিল নেই, যেখানে বলা হয়েছে:
1. (chemistry) the existence of different kinds of crystal of the same chemical compound.
2. (biology) the existence of two or more forms of individuals within the same animal species (independent of sex differences).
বাংলায় যা-ই বলা হোক, কম্পিউটার বিজ্ঞানে এ শব্দের অর্থ হলো: একই বস্তুর অনেক রূপ।

উপরের ডায়াগ্রাম থেকে স্পষ্ট, পরিমোরফিজমের স্টেটিক [স্থিত] শাখার মেথড [কোড পদ্ধতি] হলো অভারলোডিং [অতিরিক্ত বোঝাই করা]। অরপদিকে ডাইনামিক [বেগবান] শাখার মেথড হবে অভাররাইডিং [পদদলিতকরণ!]।

১. Static Polymorphism [স্থিত পলিমোরফিজম্]: একটি উদাহরণ

একে প্রাথমিক বন্ধন বা Early Binding কিংবা ভাষা পরিবর্তনকাল অর্থাৎ Compile Time Polymorphism-ও বলে। অন্যকথায় সিদ্ধান্ত নেওয়া হয় কম্পাইল টাইমে। মেথড অভারলোডিং হলো কম্পাইল টাইমেই কম্পাইলার জানতে পারে কোন্ মেথড সম্পাদিত হবে। কম্পাইল টাইমে কম্পাইলার প্যারামিটার সংখ্যা এবং প্যারামিটারের ধরন বুঝে সিদ্ধান্ত নেয় কোন্ অভারলোডেড মেথড সম্পাদিত [execute] হবে। উদাহরণ কোডটি দেখে নিই।

```
Public Class Student 'class starts
   'show method with 3 overloaded versions
   Public Overloads Function show(name As String) As String 'method with 1 parameter
      Return name
   End Function
```

[বাকি কোড পরের পৃষ্ঠায়]

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

```
    Public Overloads Function show(name As String, rollNo As String) As String 'method with 2 parameter
        Return (name + " " + rollNo)
    End Function

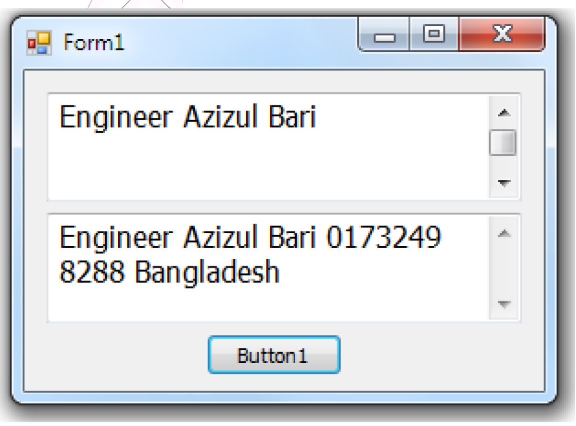
     'method with 3 parameter
    Public Overloads Function show(name As String, rollNo As String, Section As String) As String
       Dim St As String = (name + " " + rollNo + " " + Section)
       Form1.TextBox2.Text = St 'display result
        Return Nothing
    End Function
End Class

Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        TextBox1.Text = "" 'clear text
        TextBox2.Text = "" 'clear text
        CallSub()
    End Sub

    'called from button clicked event sub
    Public Sub CallSub()
        Dim s As Student = New Student()
        TextBox1.Text += (s.show("Engineer Azizul Bari")) & VbCrlf 'call show method with 1 param
        'call show method with 3 param
        TextBox1.Text += (s.show("Engineer Azizul Bari", "0173249 8288", "Bangladesh")) & VbCrlf
    End Sub
End Class
```

২. Dynamic Polymorphism [বেগবান পলিমোরফিজম]: একটি উদাহরণ

একে রান-টাইম (চলাকালীন) বা লেইট-বাইন্ডিং (দেরিতে-চলন্ত) পলিমোরফিজম-ও বলে। মেথড অভাররাইডিং এর মাধ্যমে এটা সম্পাদিত হয়। মেথড অভাররাইডিং মানে একই মেথডের বিভিন্ন রূপকে কল করা। কোন্ রূপকে কল করা হবে তা নির্ভর করবে প্রোগ্রাম চলাকালে বেগবান অবস্থায়। অন্যকথায়, রান-টাইম পলিমোরফিজমে কোন্ মেথড কল করা হবে তা নির্দিষ্ট হয় রেফারেন্স ভেরিয়েবল কর্তৃক পছন্দকৃত বস্তু থেকে। উদারহণ থেকে এসব কথার সঠিক অর্থ হয়তো অনুধাবন হতে পারে।



উপরের প্রোগ্রাম পরিচালনার ফলাফল

```
Public Class Add 'base class starts
    'Virtual method of Base Class
    Public Overridable Function Calculate(num1 As Integer, num2 As Integer) As Integer
        Form1.TextBox1.Text = ("Virtual method of base class!")
        Dim result As Integer = 0
        result = num1 + num2
        Return result
    End Function
End Class 'Base Class Ends
```

১০ম পরিচ্ছেদ: *Object Oriented Programming [OOP] -বস্তুভিত্তিক প্রোগ্রামিং*

বস্তুভিত্তিক প্রোগ্রামিং: *Object Oriented Programming [OOP]*

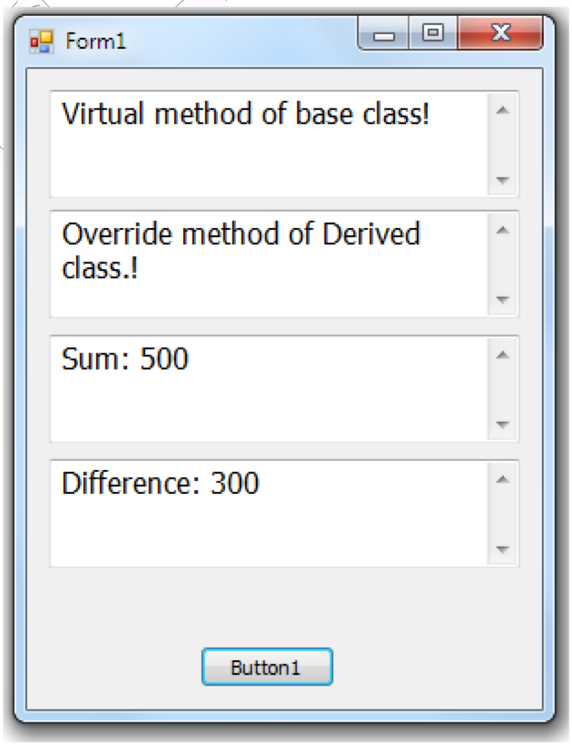
```
Public Class Subtract : Inherits Add 'Derived class starts
    'override method from Base class
   Public Overrides Function Calculate(num1 As Integer, num2 As Integer) As Integer
      Form1.TextBox2.Text = ("Override method of Derived class.!")
      Dim result As Integer = 0
      result = num1 - num2
      Return result
   End Function
End Class 'Derived class ends

Public Class Form1

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      AddMinus()
   End Sub

   'called from button clicked event sub
   Public Sub AddMinus()
      Dim a1 As Add = New Add()
      Dim AddRes As Double = (a1.Calculate(400, 100))
      TextBox4.Text = "Sum: " & AddRes
      Dim a2 As Add = New Subtract() ' instance of the derived class using base class reference
      Dim Dif As Double = (a2.Calculate(400, 100))
      TextBox3.Text = "Difference: " & Dif
   End Sub
End Class
```

দ্রষ্টব্য: এ পরিচ্ছেদে উপস্থাপিত উদাহরণগুলো মূলত অবজেক্ট অরিয়েন্টেড প্রোগ্রামিংয়ের স্বরূপ বুঝানোর জন্য দেওয়া হয়েছে। আসলে যেসব ফলাফল এগুলো থেকে পাওয়া যাচ্ছে তা অন্যভাবে অনেকটা সজহ উপায়ে প্রোগ্রাম করার মাধ্যমে অর্জন সম্ভব। আর বাস্তব ক্ষেত্রে এরূপ সাধারণ প্রোগ্রামের জন্য ব্যবহার মোটেই প্রয়োজনীয়ও নয়। যাক, আশাকরি পাঠকরা OOP-এর সাথে সম্পৃক্ত তিনটি মৌলিক বিষয় তথা, **Encapsulation, Inheritance** এবং **Polymorphism** সম্পর্কে অত্যল্প হলেও ধারণা পেয়ে গেছেন। আর এটুকু অর্জনই এ অধ্যায় রচনার উদ্দেশ্য ছিলো। আমরা OOP-নিয়ে খুব একটা মাথা ঘামাবো না। পরবর্তীতে যেসব প্রোগ্রাম উদাহরণ উপস্থাপন করবো তা হবে ভিজুয়্যাল বেসিক.নেট প্রোগ্রাম করার কৌশল বুঝা, কোড জানা এবং অভিজ্ঞতা অর্জন করার নিমিত্তে। আমরা কিন্তু OOP-ই করবো কিন্তু তা হবে আমাদের অজান্তে!



উপরের প্রোগ্রাম পরিচালনার ফলাফল

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

ভিজুয়্যাল বেসিক.নেট দ্বারা ফাইল নিয়ন্ত্রণ: VB.Net File Handling

প্রোগ্রামিং এর একটি গুরুত্বপূর্ণ অংশ হলো তথ্য বা ডাটা ফাইলে লেখা, সংরক্ষণ ও প্রয়োজনে পাঠ করা। যে উপায়ে তা করতে হয় তাকে বলে File Streaming। এর অর্থ, কম্পিউটারের মেমোরিতে সংরক্ষিত ফাইল পড়া [Read] ও লেখা [Write]। প্রত্যেক প্রোগ্রামিং ভাষায় ফাইল নিয়ে কাজ করার কোড আছে। ভিজুয়্যাল বেসিক.নেটেও ফাইল নিয়ন্ত্রণের জন্য শক্তিশালী স্ট্রিমিং সিস্টেম তৈরি করা হয়েছে। এ পরিচ্ছেদে বিভিন্ন উদাহরণসহ ফাইল নিয়ে কিভাবে কাজ করতে হয় তার ওপর আলোচনা করবো।

File- সংজ্ঞা: নামকরণকৃত মেমোরি ব্লক বা বাক্স যাতে সংরক্ষিত থাকে তথ্যাদি। কম্পিউটারের ডিস্কে [হার্ড ডিস্ক, সিডি, ভিসিডি, পেন-ড্রাইভ] সকল ফাইল প্যাথ [path] দ্বারা সনাক্ত করা হয়।
যেমন: **C:\Users\htci\Desktop\Copied\Islamic History Project**

Streaming- সংজ্ঞা: প্যাথ [রাস্তা] ব্যবহার করে কোডের মাধ্যমে ফাইল খোলা, লেখা ও পাঠ করাকে বলে স্ট্রিমিং। খোলা ফাইলকে প্রোগ্রামিং ভাষায় স্ট্রিম **[Stream]** বলে।

ইনপুট/আউটপুট [Input/Output -I/O] ক্লাস

ভিজুয়্যাল বেসিক.নেটে ফাইল নিয়ন্ত্রণের জন্য আই/অ সুযোগ-সুবিধার মূলে আছে সিস্টেমের নামকৃত [namespace] ক্লাস যাকে System.IO বলে। অর্থাৎ System.IO থেকে সব আই/অ ক্লাস উত্তরাধিকারসূত্রে প্রাপ্ত। যেসব ক্লাস আমরা ব্যবহার করতে পারবো তার মধ্যে উল্লেখযোগ্য ক'টি হলো:
System.IO.Path, System.IO.StreamReader, System.IO.StreamWriter, System.IO.StringReader এবং System.IO.StringWriter। এ পরিচ্ছেদে বিভিন্ন উদাহরণে এগুলোর ব্যবহার ও ব্যাখ্যা আসবে।

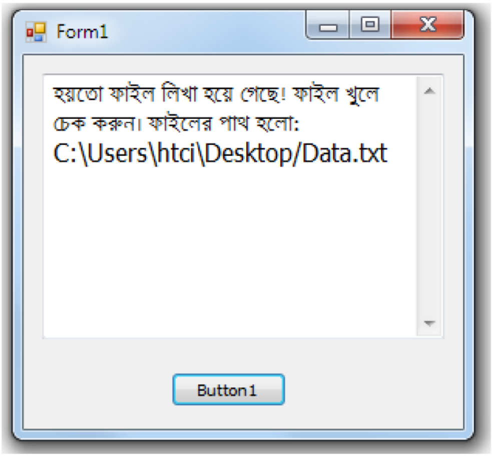
ক. স্ট্রিমরাইটার ক্লাস [StreamWriter Class]: এই ক্লাস দ্বারা খোলা ফাইলে [অর্থাৎ স্ট্রিমে] ক্যারাক্টার লেখা হয়। নিচের উদাহরণ দ্বারা বিষয়টি বুঝে আসবে।

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        WriteToFile() 'sub call
    End Sub
 'called from button clicked event sub
   Public Sub WriteToFile()
      'creating instance of StreamWriter class, and opening specific file
      Dim FOut As System.IO.StreamWriter = New System.IO.StreamWriter("C:\Users\htci\Desktop/Data.txt")
       FOut.WriteLine("বিসমিল্লাহির রাহমানির রাহীম" & vbCrLf & "ভিজুয়্যাল বেসিক.নেট একটি সহজ প্রোগ্রামিং ভাষা, অথচ খুব শক্তিশালী। আপনি অতি সহজে এ ভাষা শিখে নিতে পারেন। প্রয়োজন শুধু শেখার স্পৃহা ও ধৈর্য, এই যা!")    'writing a string to file
      FOut.Close()    'closing file -don't forget that!
       TextBox3.Text = "হয়তো ফাইল লিখা হয়ে গেছে! ফাইল খুলে চেক করুন। ফাইলের পাথ হলো: " & vbCrLf     & "C:\Users\htci\Desktop/Data.txt"
    End Sub
End Class
```

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

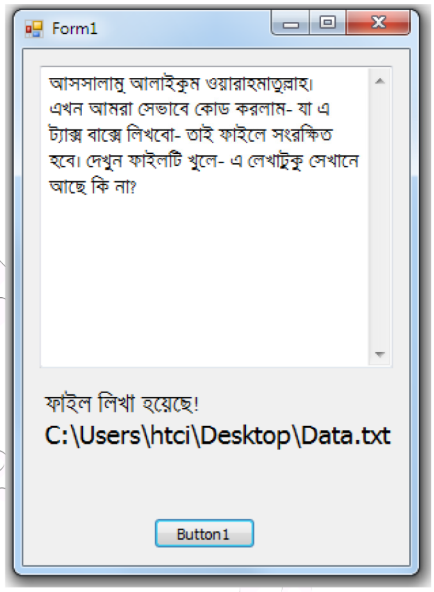
ডানের ছবিতে প্রোগ্রাম রান করার ফলাফল তুলে ধরেছি। লক্ষ করুন, বাংলা ইউনিকোড লেখা কিভাবে প্রদর্শিত হয়েছে। বিজয় কিবোর্ড বা অভ্র কিবোর্ড দিয়ে আপনি ইউনিকোড লিখতে পারেন। আরেকটি ব্যাপার এখানে উল্লেখ্য, কোডে ফাইলের প্যাথ দেওয়া আছে- কিন্তু, এই নামে প্যাথে কোনো ফাইল না থাকলে ভিজুয়্যাল বেসিক.নেট একটি ফাইল সৃষ্টি করবে। অবশ্য প্যাথের সবগুলো টেক্সট সঠিক হতে হবে।



লেখা সংরক্ষণ: আমরা চাইলে কোডটিতে সামান্য রদবদল এনে টেক্সটবাক্সে কিছু লিখে তা ফাইলে সংরক্ষণ করতে পারি। দেখুন নিচের পরিবর্তিত WriteToFile() সাবরুটিনটি।

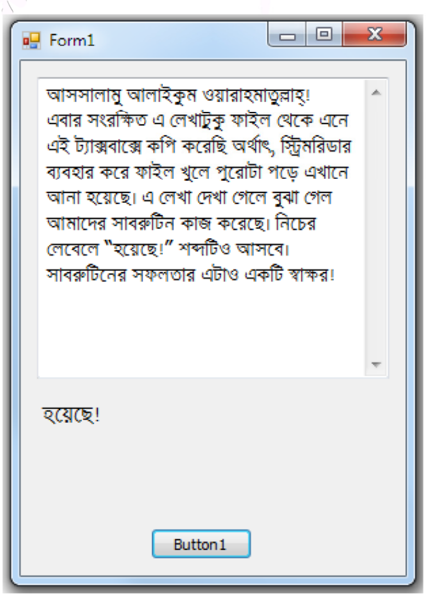
```
Public Sub WriteToFile()
      If TextBox3.Text <> "" Then 'if textBox3 dosen't have text in it nothing
will happen
          Dim WrittenText As String = TextBox3.Text 'put text into a variable
          'creating instance of StreamWriter class, and opening specific file
          Dim FOut As System.IO.StreamWriter = New
System.IO.StreamWriter("C:\Users\htci\Desktop\Data1.txt")
          FOut.WriteLine(WrittenText)
          FOut.Close()    'closing file
           Label1.Text = "ফাইল লিখা হয়েছে!" & vbCrLf &
"C:\Users\htci\Desktop\Data.txt"
      End If
    End Sub
```

এবার প্রোগ্রাম চালিয়ে টেক্সটবাক্সে কিছু লিখলে তা ফাইলে সংরক্ষিত হবে। দেখুন ডানের রান-টাইম চিত্রটি।



ক. স্ট্রিমরিডার ক্লাস [StreamReader Class]: এই ক্লাস দ্বারা খোলা ফাইল [অর্থাৎ স্ট্রিম থেকে] ক্যারাক্টার পাঠ করা হয়। নিচের উদাহরণ দ্বারা বিষয়টি বুঝে আসবে।

```
'called from button clicked event sub
    Private Sub ReadFile()
        'creating instance of StreamReader class, and opening specific file
        Dim FIn As System.IO.StreamReader = New System.IO.StreamReader
      ("C:\Users\htci\Desktop\Data1.txt")
        TextBox3.Text = FIn.ReadToEnd() 'read the whole file at once
        FIn.Close()   'closing file
         Label1.Text = "হয়েছে!"
End Sub
```



উপরের সাবরুটিন রান করার ফলে ফর্মের টেক্সটবাক্সে আসা ফাইলের লেখা ডানের ছবিতে চিত্রিত হয়েছে।

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

Using the OpenFileDialog / SaveFileDialog Class: ফা্ইল খোলার ডায়লগের ব্যবহার

ফাইল নিয়ন্ত্রণ সম্পর্কে এ পর্যন্ত আমরা যা জেনেছি তাহলো স্ট্রিমরিডার ক্লাসের মাধ্যমে ফাইল খোলা, পড়া ও প্রদর্শন। আমরা এটাও জেনেছি কিভাবে ফাইলে লিখা যায়। এজন্য প্রয়োজন স্ট্রিমরাইটার ক্লাস। তবে উভয় ক্ষেত্রেই আমাদেরকে ফাইলের প্যাথ [path] পুরোটা লিখে দিতে হয়েছে। কিন্তু আমরা সচরাচর দেখে আসছি ওয়ার্ড প্রসেসর, পেইন্ট প্রোগ্রাম ইত্যাদি থেকে ফাইল খুলতে হলে একটি ডায়লগ আসে। আমরা যে ধরনের ফাইল খুলবো তা চয়েজ করে 'OK' বোতামে ক্লিক করলেই ফাইলটি খুলে টেক্সটবাক্সে বা ক্যানভাসে প্রদর্শিত হয়। অনুরূপ নতুন ফাইল সংরক্ষণকালেও ফা্ইল সংরক্ষণের ডায়লগ আসে। এ দু'টো কোডের মাধ্যমে কিভাবে নিয়ন্ত্রণ করা হয়? এখনই আমরা জেনে নেবো।

ভিজুয়্যাল বেসিক.নেটে ফাইল খোলা ও সংরক্ষণের দু'টি ক্লাস আছে। এদের নাম যথাক্রমে: OpenFileDialog() ও SaveFileDialog() । উদাহরণের মাধ্যমে এবার এ দু'টোর ব্যবহার উপস্থাপন করবো।

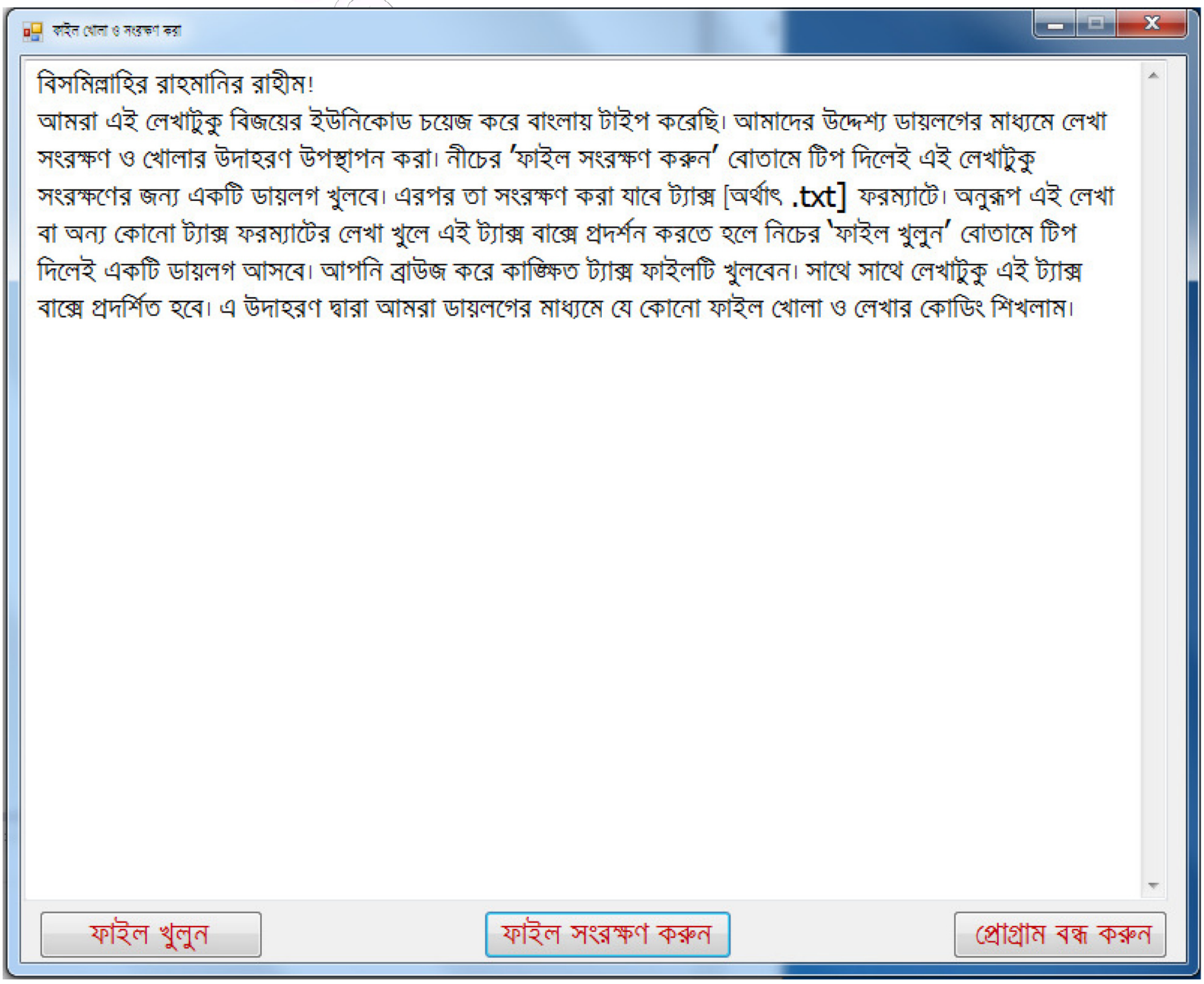
প্রথমে ফর্মের কোডটি দেখুন:

```
Public Class Form1
    'this sub clls the OpenFile Sub which in turn Opens the OpenDialog
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        OpenTheFile() 'sub call
    End Sub
    Sub OpenTheFile()   'this sub diplays open dialog
        Dim Open As New OpenFileDialog() 'new instance of file opening dialog class
        Dim myStreamReader As System.IO.StreamReader 'dimension a streamReader Object
        Open.Filter = "Text[*.txt*]|*.txt|All Files [*,*]|*.*" 'text files filters
        Open.CheckFileExists = True 'this function makes sure the file exists before opening
        Open.Title = "ফাইল খুলুন" 'open dialog title text'
        Open.ShowDialog(Me) 'shows the dialog
        Try 'Try is a debug keyword. If any problem it will catch it and jump to catch ex as exception line below
            Open.OpenFile() 'open chosen file
            myStreamReader = System.IO.File.OpenText(Open.FileName) 'assign file to StreamReader object
            TextBox1.Text = myStreamReader.ReadToEnd() 'read and display the whole file
        Catch ex As Exception 'any exception will be displayed
        End Try
    End Sub
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        If TextBox1.Text <> "" Then 'make sure there is text to be saved
            SaveTheFile() 'sub call
        End If
    End Sub
    Sub SaveTheFile() 'this dub displays Save Dialog
        Dim Save As New SaveFileDialog() 'new instance of file saving dialog class
        Dim myStreamWriter As System.IO.StreamWriter 'dimension a streamWriter Object
        Save.Filter = "Text (*.txt)|*.txt|All files(*.*)|*.*" 'text filters
        Save.CheckPathExists = True
        Save.Title = "ফাইল সংরক্ষণ করুন" 'saveDialog title text
        Save.ShowDialog(Me) 'Me means ShowDialog class itself
        Try
            myStreamWriter = System.IO.File.AppendText(Save.FileName) 'append already saved file- shows a replace
dialog
            myStreamWriter.Write(TextBox1.Text) 'write all text of the textBox to the file
            myStreamWriter.Flush() 'clear memory
        Catch ex As Exception
        End Try
    End Sub
   Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
        Close() 'close form and exit application
    End Sub
End Class
```

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

পূর্বের পৃষ্ঠার কোড রানের ফলাফল নিচে দেখানো হয়েছে। আপনি চাইলে একটি নতুন প্রজেক্ট তৈরি করতে পারেন। লক্ষ করুন, ইউনিকোড দিয়ে কিভাবে বাংলা লিখেছি। এছাড়া ইউন্ডোজের বোতামগুলোর টেক্সট প্রোপার্টিও ইউনিকোডের মাধ্যমে বাংলায় করেছি। নিজের ভাষায় এভাবে ইউজার ইন্টারফেইস তৈরি করতে পেরে অনেকটা স্বস্তি ও আনন্দবোধ করছি। মাত্র কয়েক বছর পূর্বেও এভাবে করা কঠিন ছিলো। ডিজাইনার ফন্ট [যা মূলত ইংরেজি ক্যারাক্টার] দিয়ে অতীতে কাজ করতে হয়েছে। এখন সব ইউনিকোড প্রায় সকল উইন্ডোজ সিস্টেম সাপোর্ট করে। সুতরাং বাংলায় সফটওয়্যার তৈরি অনেকটা সহজ হয়েছে। আশা করছি ভবিষ্যতে বাংলা ভাষাভাষি প্রোগ্রামাররা এই সুযোগ ব্যবহার করবেন ও সবার জন্য তৈরি করবেন মনোমুগ্ধকর মাতৃভাষা-ভিত্তিক কম্পিউটার সফট্ওয়্যার।



চিত্র ১৯: আগের পৃষ্ঠার কোড রান করার ফলাফল। লক্ষ করুন: জানালার পুরোটাই বাংলায় উপস্থাপিত হয়েছে। এজন্য আমরা বাংলা ফন্ট চয়েজ করি নি। শুধুমাত্র বিজয়ের ইউনিকোড টাইপ সিস্টেম অন করেছি। মাইক্রসফটের ভিজুয়্যাল স্টুডিও এক্সপ্রেস ফর ডেস্কটপ ইউনিকোড সাপোর্ট করে। এ কারণেই বাংলায় সবকিছু সম্ভব।

ভিজুয়্যাল বেসিক.নেটে ফাইল ম্যানেজমেন্ট সম্পর্কে আরো এক'দুটো ব্যাপার জানার আছে। পরবর্তী পৃষ্ঠাসমূহে উদাহরণসহ এগুলো তুলে ধরছি।

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

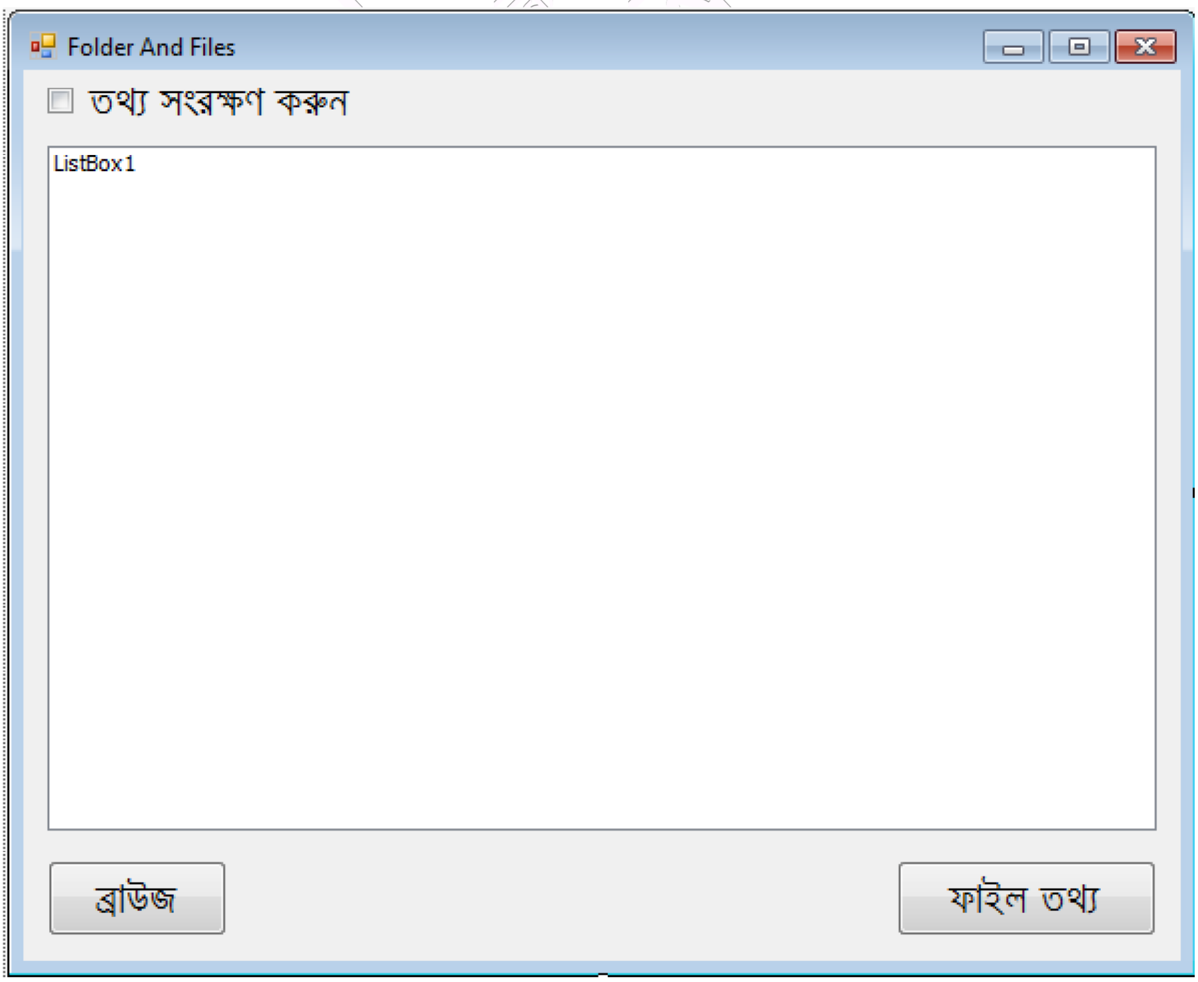
উদাহরণ-১: My.Computer.FileSystem Object ব্যবহার করে ফা্ইল নিয়ন্ত্রণ

আমরা ইতোমধ্যে ফা্ইল নিয়ন্ত্রণ সম্পর্কিত যেসব ব্যাখ্যা ও উদাহরণ তুলে ধরেছি তা সব্ই ছিলো **System.IO namespace** -এর অন্তর্ভুক্ত। ভিজুয়্যাল বেসিক.নেটে ফাইল নিয়ে কাজ করার আরো একটি পদ্ধতি আছে। একে বলে **My.Computer.FileSystem Object** । নিম্নে এ সম্পর্কিত একটি উদাহরণ তুলে ধরছি। এসাথে শেষের দিকে একই কাজ সারতে প্রথম পদ্ধতি তথা **System.IO** -এর অন্তর্ভুক্ত কোডও দেওয়া হয়েছে। শিক্ষার্থীরা উভয় কোডের মধ্যে পার্থক্য লক্ষ্য করবেন।

এবার আমরা ধারাবাহিক পদক্ষেপে পুরো উদাহরণটি লিপিবদ্ধ করবো। শিক্ষার্থীরা নিচের প্রতিটি পদক্ষেপ সঠিকভাবে অনুসরণ করে এ প্রজেক্টটি তৈরি করুন।

ক. একটি নতুন উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করুন। প্রজেক্টের নামকরণ করুন ‘FolderAndFiles’। অবশ্য চাইলে অন্য নামও দিতে পারেন। প্রদর্শিত ফর্মের প্রোপার্টি হবে নিম্নরূপ:

খ. ফর্মে টুল বাক্স থেকে সংযুক্ত করুন একটি **ListBox**, একটি **CheckBox**, দু’টি **Button** এবং একটি **FolderBrowserDialog** কন্ট্রোল। টুল বাক্সে যেয়ে এগুলোর নামের ওপর ডবল ক্লিক করলেই ফর্মে এসে যুক্ত হবে। সকল কন্ট্রোলের নাম নিচের জানালার অনুরূপ করুন। আমি বাংলা নামকরণ করেছি। চাইলে আপনি অন্য নাম বা এমনকি ইংরেজিতেও নামকরণ করতে পারেন।



১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

ফর্মের বিভিন্ন বস্তুর ব্যাখ্যা: দেখতেই পাচ্ছেন, ফর্মে মোট ৫টি বস্তু আছে: ১. লিস্টবক্স [listBox1]- এই ফিল্ডে ফল্ডারে প্রাপ্ত ডকুমেন্টের প্যাথসহ নাম প্রদর্শিত হবে। ২.'ব্রাউজ' বোতাম [Button2] এর ক্লিক ইভেন্টে আমরা কোড লিখবো ফল্ডার চয়েজ করে তথ্য সংগ্রহ করতে এবং তা লিস্ট বাক্সে নিয়ে আসতে। ৩. 'ফাইল তথ্য'বোতাম [Button1] এর ক্লিক ইভেন্টে যে কোড দেওয়া হবে, তার দ্বারা আমরা লিস্ট বাক্স সিলেক্ট করা ফাইলের ওপর পুরো তথ্যাদি সংগ্রহ করে ম্যাসেজ বাক্সে দেখাবো। ৪. 'তথ্য সংরক্ষণ করুন' নামক চেক বাক্সের [checkBox1] মাধ্যমে আমরা ফাইল তথ্য বোতামে ক্লিক করার পর তা লগ [সংরক্ষণ] করবো কি না সে সিদ্ধান্ত নেবো। ব্যবহারকারী চেক বাক্সে ক্লিক করে থাকলে বুঝা যাবে সংরক্ষণ হবে, অন্যথায় নয়। ফর্মের বাইরে লাইনের নিচে 'FolderBrowserDialog1' নামক আরেকটি বস্তু আছে। এটার কাজ হলো কোডের মাধ্যমে ডাকা ফল্ডার ব্রাউজার

> *নোট:* কোড চলাকালে যে কোনো বস্তুকে ইঙ্গিত বা পারিচিত করা হয় তার Name Prperty-এর মাধ্যমে। যেমন: আমাদের এই উদাহরণে listBox1, Button1, Button2, checkBox1 এবং Form1। তবে বস্তুর টেক্সট প্রোপার্টি জানালায় প্রদর্শিত হবে। এটা যে কোনো কিছু হতে পারে- এতে কোডের ক্রিয়ার ওপর কোনো রদবদল আনবে না। সুতরাং আপনি চাইলে উভয় বোতাম, চেক বাক্স ও ফর্মের শিরোনামের টেক্সট প্রোপার্টি অন্য কিছু দিতে পারেন। আমরা এখানে বাংলা দিয়েছি- আপনি অন্য কোনো বাংলা টেক্সট বা এমনকি ইংরেজি টেক্সটও দিতে পারবেন - এতে কোডের একটি অক্ষরও বদলানো লাগবে না।

প্রদর্শন করা। এটা ফর্মের ভেতরে থাকার বস্তু নয়। যাক, ইউ.আই. ডিজাইনের কাজ শেষ হয়ে গেলো। এবার আমরা কোড লিখবো। আমরা পুরো কোডটিই এক সঙ্গে তুলে ধরছি। এরপর প্রত্যেক সাবরুটিনের ওপর ব্যাখ্যা দেবো।

```
Public Class Form1
 'where to start browsing? Set it to current directory
 Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
   ' Set the default directory of the folder browser to the current directory.
   FolderBrowserDialog1.SelectedPath = My.Computer.FileSystem.CurrentDirectory
   SetEnabled() 'CheckBox1 and Button1 enabled
 End Sub
 'The FolderBrowserDialog1.ShowDialog call opens the Browse For Folder dialog box.
 'After the user clicks OK, the SelectedPath property is sent as an argument to the
 'ListFiles method [below].
 Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
   If FolderBrowserDialog1.ShowDialog() = DialogResult.OK Then
     ' List files in the folder.
     listFiles(FolderBrowserDialog1.SelectedPath)
   End If
   SetEnabled() 'CheckBox1 and Button1 enabled
 End Sub
 'The GetFiles method retrieves a collection of strings, one for each file in the directory.
 'The GetFiles method accepts a search pattern argument to retrieve files that match a particular pattern.
 'In this example, only files that have the extension .txt, .Doc and 'Docx [word files] are returned.
 'Returned string added to ListBox
 Private Sub ListFiles(ByVal folderPath As String)
   ListBox1.Items.Clear() 'clear any list in it
   Dim fileNames = My.Computer.FileSystem.GetFiles(folderPath, FileIO.SearchOption.SearchTopLevelOnly,
 "*.txt", "*.Doc", "*.Docx")
   For Each fileName As String In fileNames 'File names contained indexed file path
     ListBox1.Items.Add(fileName) 'build the list line by line
   Next
 End Sub
```

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    If ListBox1.SelectedItem Is Nothing Then 'make sure user has selected an item
        MessageBox.Show("Please select a file.")
        Exit Sub
    End If
   Dim filePath = ListBox1.SelectedItem.ToString ' Obtain the file path from the list box selection.
    ' Verify that the file was not removed since the '' button was clicked.
    If My.Computer.FileSystem.FileExists(filePath) = False Then
        MessageBox.Show("File Not Found: " & filePath)
        Exit Sub
    End If
    Dim fileInfoText As String = GetTextForOutput(filePath)  ' Obtain file information in a string.
    MessageBox.Show(fileInfoText) ' Show the file information.
    'The if-end if block of code sets the log file path to put it in the same directory as that of the selected file.
    'The text of the log entry is set to the current date and time followed by the file information.
    'The WriteAllText method, with the append argument set to True, is used to create the log entry.
    If CheckBox1.Checked = True Then
        ' Place the log file in the same folder as the examined file.
        Dim logFolder As String = My.Computer.FileSystem.GetFileInfo(filePath).DirectoryName
        Dim logFilePath = My.Computer.FileSystem.CombinePath(logFolder, "log.txt")
        Dim logText As String = "Logged: " & Date.Now.ToString & vbCrLf & fileInfoText & vbCrLf & vbCrLf
        ' Append text to the log file.
        My.Computer.FileSystem.WriteAllText(logFilePath, logText, append:=True)
    End If
End Sub
Private Function GetTextForOutput(ByVal filePath As String) As String
    If My.Computer.FileSystem.FileExists(filePath) = False Then
        Throw New Exception("File Not Found: " & filePath) ' Verify that the file exists. In no, show dialog & exit
    End If
    ' Create a new StringBuilder, which is used  to efficiently build strings.
    Dim sb As New System.Text.StringBuilder()
    ' Obtain file information.
    Dim thisFile As System.IO.FileInfo = My.Computer.FileSystem.GetFileInfo(filePath)
    ' Add file attributes.
    sb.Append("File: " & thisFile.FullName)
    sb.Append(vbCrLf)
    sb.Append("Modified: " & thisFile.LastWriteTime.ToString)
    sb.Append(vbCrLf)
    sb.Append("Size: " & thisFile.Length.ToString & " bytes")
    sb.Append(vbCrLf)
    ' Open the text file.
    Dim sr As System.IO.StreamReader =
        My.Computer.FileSystem.OpenTextFileReader(filePath)
    If sr.Peek() >= 0 Then
        sb.Append("First Line: " & sr.ReadLine()) ' Add the first line from the file.
    End If
    sr.Close()
    Return sb.ToString
End Function
```

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

```
  'enable or disable Button1 and CheckBox wether ListBox1 item selected or not respectively
    Private Sub SetEnabled()
        'AnySelected becomes Yes or No depending wether any Item selected or not resectively
        Dim AnySelected As Boolean = (ListBox1.SelectedItem IsNot Nothing)
        Button1.Enabled = AnySelected 'if AnySelected True, Button1 is Enabled
        CheckBox1.Enabled = AnySelected 'if AnySelected True, CheckBox1 is Enabled too
    End Sub
    Private Sub ListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
ListBox1.SelectedIndexChanged
        SetEnabled() 'CheckBox1 and Button1 enabled
    End Sub
End Class
```

কোডের ব্যাখ্যা: ফর্মে লোড ইভেন্ট হ্যান্ডলারে আমরা একটিমাত্র লাইন লিখেছি। লাইনটি এই:
FolderBrowserDialog1.SelectedPath = My.Computer.FileSystem.CurrentDirectory

এখানে ফল্ডারব্রাউজারডায়লগ ক্লাসের ডিফোল্ট প্যাথ বলে দেওয়া হয়েছে। ফাইল সিস্টেমের .CurrentDirectory যে ডাইরেক্টরী থেকে লিস্ট বাক্সে ফাইলের প্যাথ নিয়ে আসা হয়েছে সেই প্যাথের দিকেই নির্দেশ করে। অন্যকথায় প্রথমবার যে ডাইরেক্টরিতে যেয়ে আপনি ফল্ডার খুলবেন, সে-ই ডাইরেক্টরিকেই দ্বিতীয়বার দেখাবে। আর লগ টেক্সটটিও সেখানে সংরক্ষিত হবে। আমরা যদি My.Computer.FileSystem.CurrentDirectory এর বদলে এখানে **System.IO** ব্যবহার করি তাহলে লিখবো:
FolderBrowserDialog1.SelectedPath = System.IO.Directory.GetCurrentDirectory()

GetCurrentDirectory() -সিস্টেম.আইও এর একটি ফাংশন। এটা রিটার্ন করে বর্তমান ডাইরেক্টরির প্যাথ। সাবের দ্বিতীয় লাইনে অপর একটি সাবরুটিন কল করা হয়েছে। এ'টি 'ফাইল তথ্য' বোতাম ও 'তথ্য সংরক্ষণ করুন' চেক বাক্সকে অকার্যকর করে। এগুলো কার্যকর হবে এই একই সাব পুনরায় কল করে- যখন ব্যবহারকারী কোনো ফাইলের তথ্য জানার জন্য লিস্ট বাক্সের কোনো আইটেমে ক্লিক করবে।

Button2 -এর ক্লিক ইভেন্ট সাবে; If FolderBrowserDialog1.ShowDialog() = DialogResult.OK Then -অংশ প্রথমে ডায়লগটি দেখাবে। এরপর ব্যবহারকারী কোনো ফল্ডার চয়েজ করে 'OK' বোতামে ক্লিক করলে বুলিয়ান ফলাফল 'True' হবে। সুতরাং পরের লাইনে যেয়ে ListFile() সাবটি কল করবে- যার কাজ প্রাপ্ত ফা্ইলগুলোর নাম লিস্ট বাক্সে প্রদর্শন করা। আর যদি 'Cancel' বোতামে ক্লিক করা হয় তাহলে কিছুই হবে না। শুধুমাত্র ডায়লগটি বন্ধ হবে। একই সময় SetEnabled() সাব কল করে নিশ্চিত করা হবে 'ফাইল তথ্য' বোতাম ও তথ্য 'সংরক্ষণ করুন' চেক বাক্স যাতে Disabled অবস্থায় থাকে। অবশ্য ডায়লগের ফলাফল 'OK' কিংবা 'Cancel' হোক উভয় অবস্থায়ই Button2-তে ক্লিক করলে এ দু'টো Disabled থাকা জরুরী- আর তা-ই হবে। কারণ আমরা কল করছি **End If** কথাটির পরে।

ListFiles(ByVal folderPath As String) সাবের কাজ হলো প্রদত্ত ফল্ডার প্যাথ [সাবের আর্গুমেন্ট] ব্যবহার করে ঐ ফল্ডানের .txt, .Doc & .Docx ফাইল টাইপের যতো ফাইল আছে সবগুলোর প্যাথ সংগ্রহ করে লিস্ট বাক্সে প্রদর্শন করা। এজন্য অপারেটিং সিস্টেমের My.Computer.FileSystem.GetFiles ফাংশনটি কাজে লাগানো হয়েছে। এ ফাংশনের গুরুত্বপূর্ণ দু'টি আর্গুমেন্ট হলো: ফল্ডারের প্যাথ [যা আসবে FolderBrowserDialog1.SelectedPath থেকে] ও অনুসন্ধান ফিল্টার তথা [এখানে] .txt, .Doc & .Docx ফাইল টাইপ। অপর আর্গুমেন্ট নিশ্চিত করে শুধুমাত্র উচ্চ পর্যায়ের ডাইরেক্টরি থেকে তথ্য আনা হবে। অন্যকথায় অনুসন্ধান হবে শুধুমাত্র কারেন্ট ফল্ডারে- এর ভেতরস্থ অপর কোনো ফল্ডার এ অনুসন্ধানের আওতাভুক্ত থাকবে না।

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*

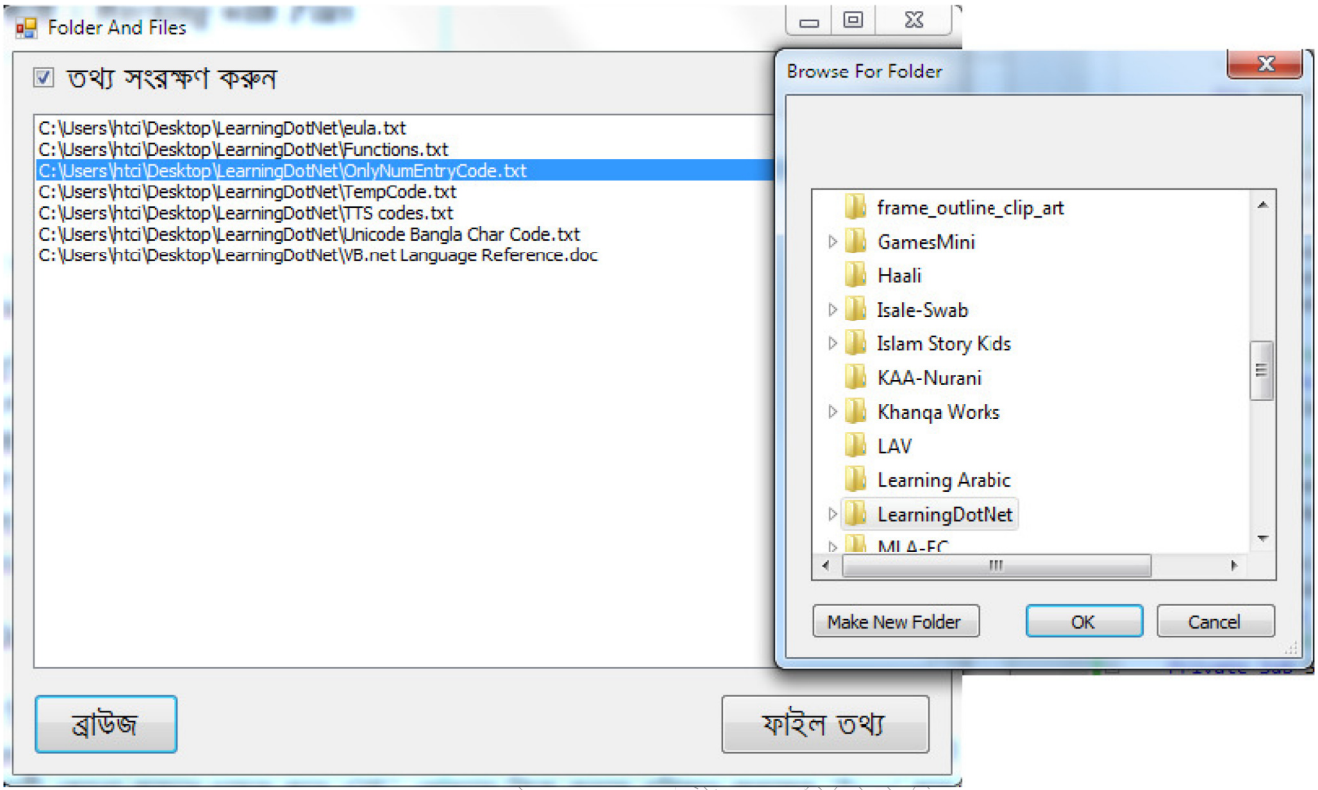
লিস্ট বাক্সে আইটেমগুলো লাইনের পর লাইনে লেখা হয় For Each - Next কোড অংশের মাধ্যমে।

Button1_Click ইভেন্ট সাবের কোডগুলো লিস্ট বাক্সের সিলেক্টেড ফাইল প্যাথে সংরক্ষিত ফাইলের তথ্যাদি একটি লগে [অর্থাৎ টেক্সট ফাইলে] সংরক্ষণ করবে কারেন্ট ডাইরেক্টরির ভেতর। প্রত্যেক বার কোনো ফাইলের প্যাথ সিলেক্ট করে এই বোতামে ক্লিক করলেই ঐ লগে তথ্যাদি যোগ হবে। সুতরাং লগ ফাইল একটি মাত্র তৈরি করা হবে।
সাবের প্রথম If-End If ব্লক নিশ্চিত করে লিস্ট বাক্সের কোনো আইটেম সিলেক্ট করা আছে কি না। না থাকলে মেসেজ জানালা এসে 'Please select a file.' কথাটি জানিয়ে কোডের কার্যক্রম এখানে শেষ হবে।
পরের লাইনে আমরা ফাইল প্যাথের লেখাটুকু একটি ভেরিয়েবলে সংরক্ষণ করেছি। এরপর আরেকটি If-End If ব্লকে নিশ্চিত করা হচ্ছে ফাইলটি আদৌ আছে কি না- কারণ, ইতোমধ্যে এটি মুছে যেতে পারে কিংবা অন্যত্র সরিয়ে নিতে পারেন ব্যবহারকারী। পরের লাইনে ফাইলের তথ্যাদি একটি ভেরিয়েবলে সংরক্ষণ করা হচ্ছে। এই তথ্যাদি আসবে আমাদের বানানো GetTextForOutput(filePath) নামক সাব থেকে। এ সাব সম্পর্কে একটু পরই ব্যাখ্যা তুলে ধরবো। পরের লাইনে মেসেজ বাক্সে প্রাপ্ত তথ্যাদি দেখানো হবে। এরপর আরেকটি If-End If ব্লকের ভেতর তথ্যাদি সংরক্ষণ করার জন্য লগ ফাইল বানানো হওয়ার কাজ সারা হচ্ছে। যদি চেক বাক্স ক্লিক করা থাকে তাহলেই এ কাজ হবে।
তথ্য সংরক্ষণের জন্য প্রথমে ডাইরেক্টরি তৈরি করে তা logFolder নামক ভেরিয়েবলে প্রাপ্ত টেক্সট রাখা হয়েছে। এ কাজের জন্য সিস্টেমের My.Computer.FileSystem.GetFileInfo(filePath).DirectoryName ক্লাসটি কল করেছি। এর পরের লাইনে লগ ফাইল সৃষ্টি হয়েছে। এর প্যাথটিও আমার logFilePath নামক ভেরিয়েবলে সংরক্ষণ করেছি। পরের লাইনে লগে কি লেখা হবে তা-ও নাকটি logText নামক ভেরিয়েবলে সংরক্ষণ করে লেখার কাজটি সেরেছি এই My.Computer.FileSystem.WriteAllText(logFilePath, logText, append:=True) কোড লাইনটি দ্বারা।
পরবর্তী তিনটি ব্লকের প্রথমটি ফাংশন এবং বাকি দু'টো সাবরুটিন। এগুলো প্রোগ্রামের অন্যত্র থেকে কল করা হয়।
ফাংশন GetTextForOutput কাজ হলো ফাইল প্যাথ [filePath] প্যারামিটার ব্যবহার করে ফা্ইল সম্পর্কে তথ্য সংগ্রহ করা ও তা টেক্সট হিসাবে কলিং কোডে ফিরে দেওয়া। এ কাজ আঞ্জাম দিতে ফাংশন ধারাবাহিকভাবে সিস্টেমের কয়েকটি গুরুত্বপূর্ণ মেথড কাজে লাগায়। যেমন:
১. My.Computer.FileSystem.FileExists(filePath) -এটা রিটার্ন করে একটি বুলিয়ান ভ্যালু। ফা্ইলের অস্তিত্ব থাকলে তা হবে True, অন্যথায় False।
2. New System.Text.StringBuilder() -এটা একটি সিস্টেম অবজেক্ট মেথড। এর কাজ হলো ফাইল সম্পর্কে বিভিন্ন তথ্যাদি যোগান দেওয়া: যেমন, ফাইলের পূর্ণাঙ্গ নাম, সর্বশেষ রদবদলের সময়, ফাইলের বাইট্স সংখ্যা [সাইজ] ইত্যাদি।
৩. My.Computer.FileSystem.GetFileInfo(filePath) -এই ফাংশন মেথড কল করে আমরা ফাইলের তথ্যাদির প্রতি পয়েন্ট করতে পারি।
৪. System.IO.StreamReader এবং My.Computer.FileSystem.OpenTextFileReader(filePath) ব্যবহার করে ফাইল খুলে পরের ব্লকে প্রথম লাইন পড়া এবং ফা্ইল বন্ধ করা।

এর পরের কোড ব্লক হলো লিস্ট বাক্সের SelectedIndexChanged ইভেন্ট হ্যান্ডলার ধরে SetEnabled সাবটি কল করা। আর এই সর্বশেষ সাবটির কাজ হলো 'তথ্য সংরক্ষণ করুন' চেক বাক্স ও 'ফাইল তথ্য' বোতামের Enabled প্রপার্টি সত্য হিসাবে নিশ্চিত করা। ব্যস্! এভাবেই কোডটি কাজ করে! পাঠকদের নিকট এই ব্যাখ্যা প্রথমত কঠিন মনে হবে। ভাববেন না, ধীরে ধীরে আপনি বুঝতে পারবেন ঠিকই। ইংরেজিতে একটি কথা আছে: Practice makes perfect। এ কথাটি ও নিজের বুঝার ক্ষমতার প্রতি আস্থা রাখুন। এ গ্রন্থের বিভিন্ন কোড উদাহরণ, প্রজেক্ট ইত্যাদি প্রয়োজনে বার বার স্টাডি করুন। আপনি অভিজ্ঞ প্রোগ্রামার বনতে মোটেই ব্যর্থ হবেন না। পরের পৃষ্ঠায় ফাইল নিয়ন্ত্রণের এই উদাহরণ প্রজেক্টটির রান-টাইম জানালা তুলে ধরেছি। একটু দেখে নিন।

১১তম পরিচ্ছেদ: *Working with Files*

ফাইল নিয়ে কাজ : *Working with Files*



উদাহরণ-১ -এর রানটাইম দু'টি জানালা।

ভিজুয়্যাল বেসিক.নেটে ফা্ইল নিয়ে কাজ করার উপর উপরোক্ত উদাহরণটি শিক্ষার্থীরা বার বার দেখে ও নিজে প্রজেক্টটি তৈরি করে ভালো করে অনুধাবনের চেষ্টা করবেন। আমরা এ্ই কোড দ্বারা দু'টি সিস্টেম ফাংশন ক্লাস সম্পর্কে অবগত হয়েছি। এগুলো হলো:

১. System.IO.File - যার ব্যবহার পরিচ্ছেদের প্রথম উদাহরণে দেখেছি।
২. My.Computer.FileSystem - যার ব্যবহার আমরা শেষের উদাহরণে দেখেছি।

উপরোক্ত উভয় ক্লাসের কাজ একই। আপনি যে কোনোটি বা উভয়টি একই এ্যাপলিকেশনে ব্যবহার করতে পারেন। তবে আমার ইচ্ছে হচ্ছে দ্বিতীয়টির ব্যবহারের উপর আলাদা একটি পরিচ্ছেদ রচনা করার। এর কার্যাকারিতা অনেক ব্যাপক। যেটুকু সম্ভব এ ক্লাস সম্বন্ধে জানা হবে বিশেষ উপকারী। তাই আসুন, পরের পরিচ্ছেদে এর ব্যবহারের আরো কিছু উদাহরণ দেখে নিই।

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

My.Computer.Object

সিস্টেমের এই বস্তুটি আমাদেরকে সুযোগ দিয়েছে কম্পিউটারের বিভিন্ন প্রোপার্টিকে ব্যবহার ও নিয়ন্ত্রণ করার। যেমন: অডিও, ক্লিপবোড, ঘড়ি, ফাইল সিস্টেম, তথ্য, কিবোর্ড, মাউস, নেটওয়ার্ক, পোর্ট, রেজিস্ট্রি ইত্যাদি। এসব নিয়ন্ত্রণের কিছু উদাহরণ এবার উপস্থাপন করছি।

১. **My.Computer.Audio:** নিশ্চয়ই আপনি লক্ষ্য করেছেন, কোনো কোনো প্রোগ্রামের মধ্যে অডিও প্লে করতে। ভিজুয়্যাল বেসিক.নেটে এটা যে কোড দ্বারা করা হয় তা-ই হলো এই বস্তুটি। নিচের উদাহরণ দ্বারা ব্যবহার পদ্ধতি আমার শিখে নেবো।

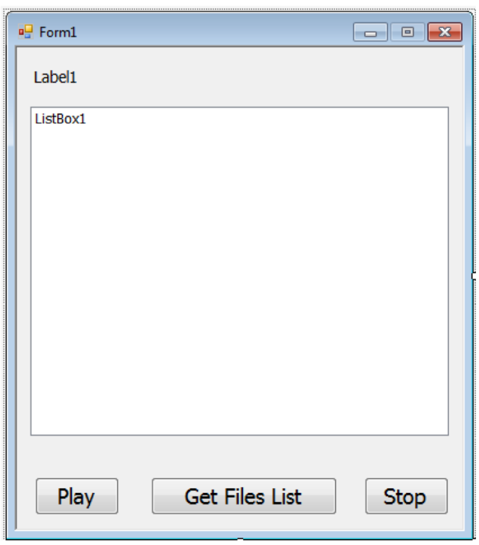
মনে করুন আমরা আল-কুরআনের একটি সূরা শুনবো। সূরাটির অডিও ফাইল সংরক্ষিত আছে যেখানে তার প্যাথ হলো: C:\Users\htci\Music\Muslima - Nas.mp3। আমরা এই ফাইলটি প্লে করতে পারি এই কোডটি দ্বারা:

```
Sub PlayBackgroundSoundFile()
    My.Computer.Audio.Play("C:\Users\htci\Music\Muslima - Nas.mp3",
AudioPlayMode.WaitToComplete) 'শেষের এই প্যারামিটার নিশ্চিত করে ফাইল প্লে শেষ না হওয়া পর্যন্ত
                        'পরবর্তী কোড রান করবে না।
End Sub
```

উপরোক্ত কোড ব্যবহার করে আমরা তৈরি করতে পারি একটি সাধারণ অডিও প্লেয়ার প্রোগ্রাম। আসুন তাহলে, এটা করে নিই।

উদাহরণ-১: Simple .wav audio player -.wav ফা্ইল অডিও প্লেয়ার

আপনার সকল অডিও ফাইলগুলো যে কোনো একটি ফল্ডারে সংরক্ষণ করে রাখতে পারেন। মনে রাখবেন ফাইলগুলো হতে হবে `.wav` ফাইল। আপনার অডিও ফাইল এই ফরমেটে না থাকলে ইন্টারনেট থেকে ফ্রিওয়্যার অডিও কনভার্টার ফাইল ডাউনলোড করতে পারেন। বেশ কয়েকটি ফ্রিওয়্যার আছে। আমি ব্যবহার করি- 'frea:ac - free audio converter v 1.0.16'। যাক, নিচের পদক্ষেপগুলো একে একে অনুসরণ করে প্রজেক্ট তৈরি করুন।



১. নতুন একটি উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করুন- এর নামকরণ করুন: wavAudioPlayer।

২. ফর্মটির ডিজাইন পাশের ছবিতে প্রদর্শিত হয়েছে। এতে থাকবে একটি লেবেল, একটি লিস্ট বাক্স ও ৩টি বোতাম।

৩. Button1 ডবল ক্লিক করুন। ফর্মের ক্লাসের ভেতর বোতামের ক্লিক ইভেন্ট সাব আসবে। এতে লিখুন নিম্নের কোডটুকু।

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    If ListBox1.SelectedItem = "" Then Exit Sub 'make sure a file path is
chosen
    PlayFile() 'call sound play sub
End Sub
```

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

প্রথম লাইন দ্বারা নিশ্চিত করা হয়েছে যে, ব্যবহারকারী লিস্ট বাক্সের কোনো একটি লাইনে ক্লিক করে সিলেক্ট করেছেন। এটা নিশ্চিত হওয়ার পর পরের লাইনে আমরা নিম্নের সাবটি কল করেছি।

```
Sub PlayFile()
    'get the path string from list box selected line
    Dim FileP As String = ListBox1.SelectedItem
    My.Computer.Audio.Play(FileP) 'play file
    Label1.Text = "Now Playing: " & FileName(FileP) 'call FileName() sub show info
End Sub
```

উক্ত সাবের প্রথম লাইনে FileP নামক স্ট্রিং ভেরিয়েবলের মধ্যে লিস্ট বাক্সের ফাইল প্যাথ স্ট্রিংটি সংরক্ষণ করেছি। এরপর সিস্টেমের My.Computer.Audio.Play ফাংশনটি কল করেছি ও এর প্যারামিটার হিসাবে FileP -অর্থাৎ ফাইলের প্যাথ দিয়েছি। সুতরাং .wav ফাইলটি খুলে তা বাজতে শুরু হবে। পরের লাইনটি যে ফাইল প্লে হচ্ছে তার নাম লেবেলে দেখাবে। এবার লিখুন বোতাম২ এর ক্লিক ইভেন্ট সাবে নিচের কোড লাইনটি। এই কোড লাইন বলছে ফাইল প্লে করা বন্ধ করতে।

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    My.Computer.Audio.Stop() 'stops playing
End Sub
```

বোতাম৩ এর ক্লিক ইভেন্টে যে কোডটি দেওয়া হয়েছে তার দ্বারা উদ্দেশ্য হলো, ফন্ডার ব্রাউজার ডায়লগ খোলা ও যে কোনো ফন্ডারে প্রাপ্ত সকল .wav সাউন্ড ফাইলের প্যাথ স্ট্রিং এনে লিস্ট বাক্সে প্রদর্শন করা। সুতরাং নিচের কোড ব্লকটি বোতাম৩ এর ইভেন্ট হ্যান্ডলারে লিখতে হবে:

```
Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
    If FolderBrowserDialog1.ShowDialog() = DialogResult.OK Then 'if Cancel, the code will not execute
        'List files in the folder.
        ListFiles(FolderBrowserDialog1.SelectedPath) 'call the sub below
    End If
End Sub
```

প্রথম লাইনের '=' চিহ্নের আগের অংশ ফন্ডার ব্রাউজার ডায়লগটি খুলবে। এরপর ফলাফল যদি 'OK' হয়- অর্থাৎ ব্যবহারকারী কোনো ফন্ডার খুলেন তাহলে পরবর্তী লাইনের কোড রান করবে- অন্যথায় নয়। আমাদের তৈরি ListFiles() সাবরুটিনটি একটিমাত্র প্যারামিটার গ্রহণ করে- সেটা হলো সংরক্ষিত ফাইলের প্যাথ- যা আসবে ফন্ডার ব্রাউজার থেকে। এই সাবরুটিনটি নিম্নরূপ:

```
'opens all .wav files in a folder and list their path in the list Box
Private Sub ListFiles(ByVal folderPath As String)
      ListBox1.Items.Clear() 'clear any previous list
      Dim fileNames As String() = System.IO.Directory.GetFiles(folderPath, "*.wav", _
      System.IO.SearchOption.TopDirectoryOnly) 'this line gets the path list
      For Each fileName As String In fileNames 'this one puts the list line by line in the ListBox
         ListBox1.Items.Add(fileName)
      Next
End Sub
```

এই সাবরুটিনের প্রথম লাইন আগের কোনো লেখা থাকলে তা লিস্ট বাক্স থেকে মুছে ফেলে। দ্বিতীয় লাইন ফন্ডার ব্রাউজার থেকে সিলেক্ট করা ফন্ডারের ভেতর যতো .wav টাইপের সাউন্ড ফাইল আছে সবগুলোর প্যাথ সংগ্রহ করে। আর For Each -- Next লুপটির কাজ হলো সবগুলো ফাইলের প্যাথ স্ট্রিং, লাইনে লাইনে লিস্টবাক্সে লিখে দেওয়া। সর্বশেষ আমাদের বানানো ফাংশনটির ব্যবহার মূলত জরুরি নয়। তবে আমরা তো শিখবো! প্যাথ স্ট্রিং থেকে শুধু ফাইলের নাম কিভাবে বের করা যায় তা-ই দেখানো হলো এ ফাংশন তৈরির উদ্দেশ্যে।

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

প্রথমে ফাংশনটি দেখে নিই। এরপর এর ব্যাখ্যা আসবে।

```
'this function returns the neame of a file from path string parameter
Private Function FileName(SupplyString As String) As String
    Dim A As Integer : Dim RetString As String = ""
    For A = Len(SupplyString) To 1 Step -1
        If Mid(SupplyString, A, 1) = "\" Then Exit For 'last '\' char position
    Next
    RetString = Mid(SupplyString, (A + 1), Len(SupplyString) - A) 'get the name
    Return RetString
End Function
```

ফাংশনের ব্যাখ্যা: প্রত্যেক প্যাথ নামের শেষের অংশে '\' -এ চিহ্নটি পরে ফাইলের নাম থাকে। সুতরাং আমরা চাই ফাংশনের প্যারামিটার হবে প্যাথ নামটি যার টাইপ হলো স্ট্রিং। ফাংশন থেকে যে ভ্যালু রিটার্ন হবে তা-ও একটি স্ট্রিং টাইপ। এজন্য আমরা প্রথম লাইনে লিখেছি:

```
Private Function FileName(SupplyString As String) As String
```

উক্ত লাইনের প্রথম দু'টি শব্দ লিখা বাধ্যতামূলক। এরপর এসেছে ফাংশনের নাম- এখানে FileName। বন্ধনীর ভেতর থাকবে প্যারামিটার- বা আর্গুমেন্ট যা কলিং কোডে দিতে হবে। কী দিতে হবে, তা আমরা সাব্যস্ত করে দিয়েছি- সেটা হলো স্ট্রিং টাইপ। এজন্য আমরা লিখেছি: (SupplyString As String)। এ তিনটি শব্দের প্রথমটি আমাদের বানানো, দ্বিতীয় ও তৃতীয়টি বাধ্যতামূলক- এখানে As String। কারণ, আমরা চাই SupplyString -এর টাইপ হবে স্ট্রিং। লাইনের শেষের দু'টি শব্দ দ্বারা আমরা জানিয়ে দিচ্ছি ফাংশন কোন্ টাইপের ভ্যালু রিটার্ন করবে- তা-হবে, স্ট্রিং। ফাংশনের দ্বিতীয় লাইনে দু'টি ভেরিয়েবল ঘোষণা করেছি: A - একটি ইন্টেজার টাইপ ও RetString -একটি স্ট্রিং টাইপ। শেষোক্তটি শূন্য স্ট্রিং হিসাবে ইনিশিয়েলাইজ করেছি। প্রথমটি পরবর্তী ফর-নেক্সট্ লুপটির জন্য ব্যবহৃত হয়েছে। এ ফর-নেক্সট্ লুপের প্রথম লাইন দ্বারা বুঝাচ্ছি, SupplyString তথা প্যাথ নামের দৈর্ঘ্যসংখ্যা থেকে ১ পর্যন্ত এই লুপটি রান করবে। আমাদের উদ্দেশ্য, ঠিক কোন্ পজিশনে স্ট্রিংয়ের মধ্যে শেষের '\' -এ চিহ্নটি আছে তা নির্ণয় করা। সুতরাং আমরা ডান থেকে শুরু করে একে একে বামদিকে যেয়ে উক্ত চিহ্নটির পজিশন খুঁজে দেখবো। যে মুহূর্তে এটা পাওয়া যাবে তখনই লুপ থেকে বের হবো। আর এ চিহ্নটি বার বার পরীক্ষার জন্য আমরা পরের লাইনের কোডটি ব্যবহার করেছি। তাহলো:

`If Mid(SupplyString, A, 1) = "\" Then Exit For` -এখানে, Mid ফাংশনের কাজ হলো স্ট্রিংয়ের পজিশন A -এর একটি মাত্র অক্ষর সংগ্রহ করা। আমরা বলছি এই অক্ষর বা ক্যারাক্টারটি যদি '\' হয়, তাহলে লুপ থেকে বের হও। মনে রাখবেন আমাদের ইন্টেজার ভেরিয়েবল A শুরু হয়েছে প্যাথ স্ট্রিং এর মোট অক্ষরসংখ্যা থেকে। আর এ সংখ্যার মান প্রত্যেক লুপে এক এক করে কমে আসছে। সুতরাং উক্ত '\' -ক্যারাক্টারটি যেখানে পাওয়া যাবে সেখানকার সংখ্যা-মান হবে A। অর্থাৎ ঠিক কোন্ সর্বডানের পজিশনে '\' -এ চিহ্নটি আছে তার মান আমরা পেলাম। এই মানটি বের করাই হলো ফাংশনের মূল কাজ।

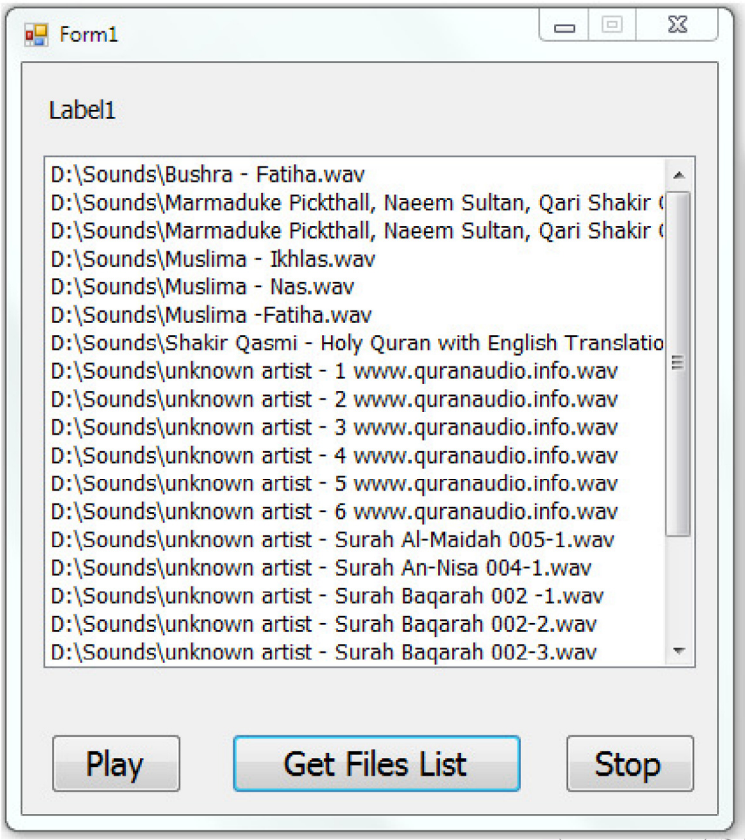
এবার আমরা ফাইলের নাম অংশ একটিমাত্র কোড লাইনে সংগ্রহ করতে পারি:

`RetString = Mid(SupplyString, (A + 1), Len(SupplyString) - A)` -এখানে Mid ফাংশনের কাজ হলো পুরো স্ট্রিং থেকে প্রাপ্ত সংখ্যা (A+1)-এ শুরু করে স্ট্রিং এর শেষ পর্যন্ত [Len(SupplyString) - A] সংগ্রহ করা। আমরা একে [অর্থাৎ ফাইলের নামকে] সংরক্ষণ করেছি RetString ভেরিয়েবলে। পরের লাইনে ফাংশন তার রিটার্ন ভ্যালু ফিরিয়ে দেবে। তাহলো এই সংরক্ষিত RetString ভেরিয়েবল: `Return RetString`।

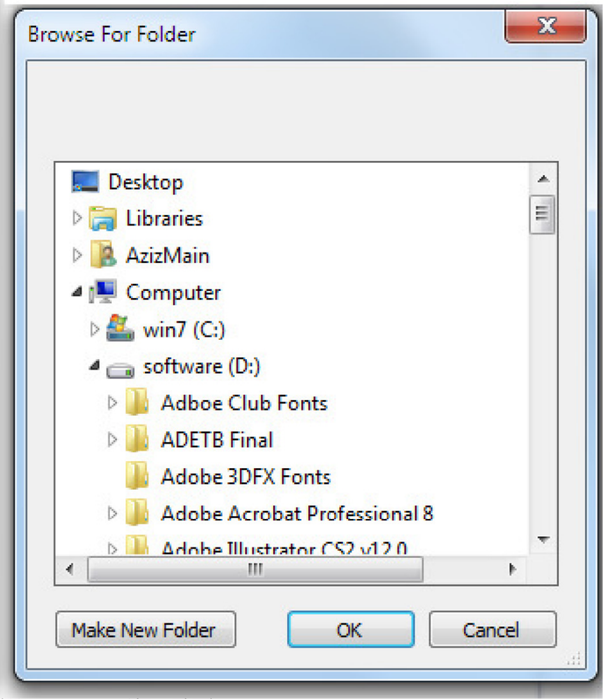
যাক, আশাকরি ফাংশনের ওপর এই বিস্তারিত ব্যাখ্যা আপনার জন্য ফাংশন কিভাবে কাজ করে তা বুঝতে সহায়ক হবে। পরের পৃষ্ঠায় একটি চিত্রে এ উদাহরণ প্রোগ্রামটির রান-টাইমের ছবি তুলে ধরেছি।

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*



'ফাইল অডিও প্লেয়ার' উদাহরণের প্রোগ্রাম চলাকালের চিত্র।



উদাহরণ-২: Me.Computer -এর বিভিন্ন ফাংশন, মেথড ও প্রোপার্টি

Me.Computer-এর বিভিন্ন উদাহরণ একত্রে একটি প্রজেক্টে তুলে ধরেছি। পাঠকদের উচিৎ একটি উইন্ডোজ ফর্ম প্রজেক্ট সৃষ্টি করে নিচের পদক্ষেপ মুতাবিক এটি ধীরে ধীরে ডেভেলাপ করা।

(ক) My.Computer.Info.AvailablePhysicalMemory এটা একটি রিড-অনলি [শুধু পাঠ্য তথ্য] প্রোপার্টি। কম্পিউটারে কি পরিমাণ বাস্তব মুক্ত মেমোরি আছে তা এই প্রোপার্টির মাধ্যমে জানা যায়।

(খ) My.Computer.Info.TotalPhysicalMemory এটা মোট বাস্তব মেমোরি কত রিটার্ন করে।

(গ) My.Computer.Info.AvailableVirtualMemory এটা রিটার্ন করে মুক্ত ভারচুয়্যাল মেমোরির মাত্রা।

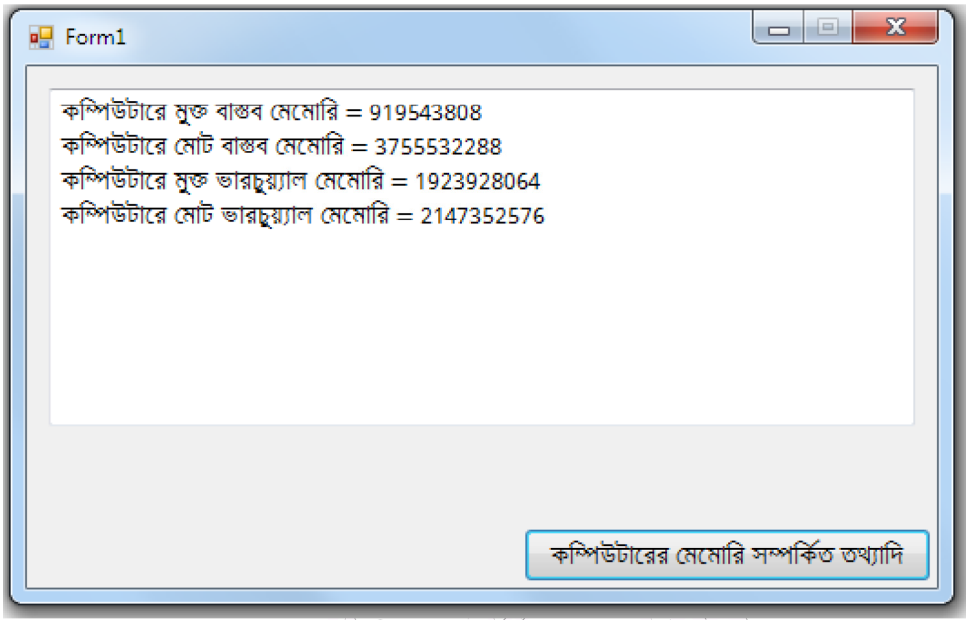
(ঘ) My.Computer.Info.TotalVirtualMemory এটা থেকে জানা যায় মোট ভারচুয়্যাল মেমোরি কত।

উপরোক্ত প্রোপার্টিগুলো কিভাবে কাজ করে দেখার জন্য প্রথমে প্রজেক্টের জানালায় একটি বোতাম ও একটি টেক্সট বাক্স যুক্ত করুন। এরপর বোতামের ক্লিক ইভেন্টে [বোতামে ডবল ক্লিক করলেই কোড জানালায় ক্লিক ইভেন্ট সাব আসবে] নিচের কোড ব্লকটি লিখুন।

```
    'information on the memories of user's computer in bytes.
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    TextBox1.Text = "" 'clear any text
    TextBox1.Text += ("কম্পিউটারে মুক্ত বাস্তব মেমোরি = " & _
    My.Computer.Info.AvailablePhysicalMemory) & vbCrLf
    TextBox1.Text += ("কম্পিউটারে মোট বাস্তব মেমোরি = " & _
    My.Computer.Info.TotalPhysicalMemory) & vbCrLf
    TextBox1.Text += ("কম্পিউটারে মুক্ত ভারচুয়্যাল মেমোরি = " & _
    My.Computer.Info.AvailableVirtualMemory) & vbCrLf
    TextBox1.Text += ("কম্পিউটারে মোট ভারচুয়্যাল মেমোরি = " & _
    My.Computer.Info.TotalVirtualMemory)
End Sub
```

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*



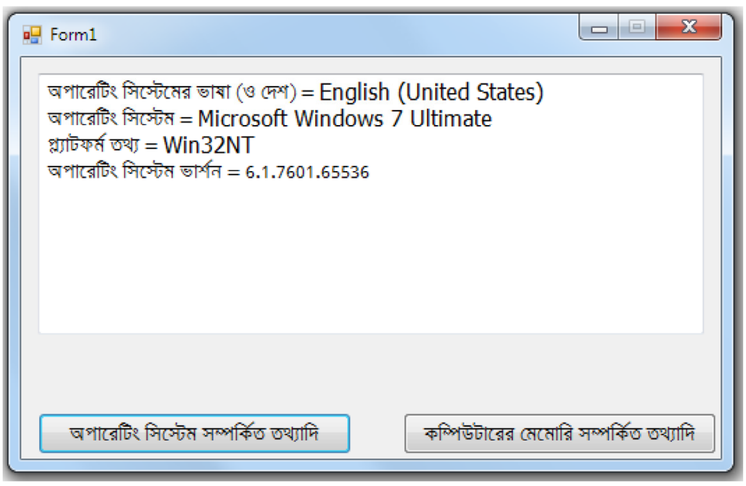
আগের পৃষ্ঠার কোড ব্লক 'কম্পিউটারের মেমোরি সম্পর্কিত তথ্যাদি' বোতামের ক্লিক ইভেন্টে ক্লিক করে রান করার ফলাফল।

(ঙ) My.Computer.Info.InstalledUICulture.DisplayName এটা দ্বারা কম্পিউটারের অপারেটিং সিস্টেমের ভাষা ও কোন্ দেশের ভাষা জানা যায়।
(চ) My.Computer.Info.OSFullName এটা অপারেটিং সিস্টেমের পূর্ণাঙ্গ নাম রিটার্ন করে।
(ছ) My.Computer.Info.OSPlatform এটা দ্বারা অপারেটিং সিস্টেমের প্ল্যাটফর্ম কি তা জানা যায়।
(জ) My.Computer.Info.OSVersion এটা দিয়ে আমরা জানতে পারি অপারেটিং সিস্টেমের ভার্শন কোন্‌টি।

উপরোক্ত প্রতিটি তথ্য জানতে ফর্মে আরেকটি বোতাম যুক্ত করুন। এর ক্লিক ইভেন্টে লিখুন:

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    TextBox1.Text = "" 'clear any text
    TextBox1.Text += ("অপারেটিং সিস্টেমের ভাষা (ও দেশ) = " & _
   My.Computer.Info.InstalledUICulture.DisplayName) & vbCrLf
    TextBox1.Text += ("অপরেটিং সিস্টেম = " & _
    My.Computer.Info.OSFullName) & vbCrLf
    TextBox1.Text += ("প্ল্যাটফর্ম তথ্য = " & _
    My.Computer.Info.OSPlatform) & vbCrLf
    TextBox1.Text += ("অপারেটিং সিস্টেম ভার্শন = " & _
    My.Computer.Info.OSVersion)
End Sub
```

লক্ষ্যণীয় যে, একই লাইনে দীর্ঘ কোড না লিখে আমরা (_) আন্ডারস্কোর দ্বারা লাইনটি ভাগ করে একাধিক লাইনে লিখতে পারি। উপরের কোডে চারটি লাইন এভাবে ভাগ করে জোড়া জোড়া লাইনে লিখেছি। আন্ডারস্কোর থাকতে হবে লাইনের শেষে। যাক, উপরের কোড রান করার ফলাফল ডানের চিত্রে তুলে ধরেছি। একটু দেখে নিন।



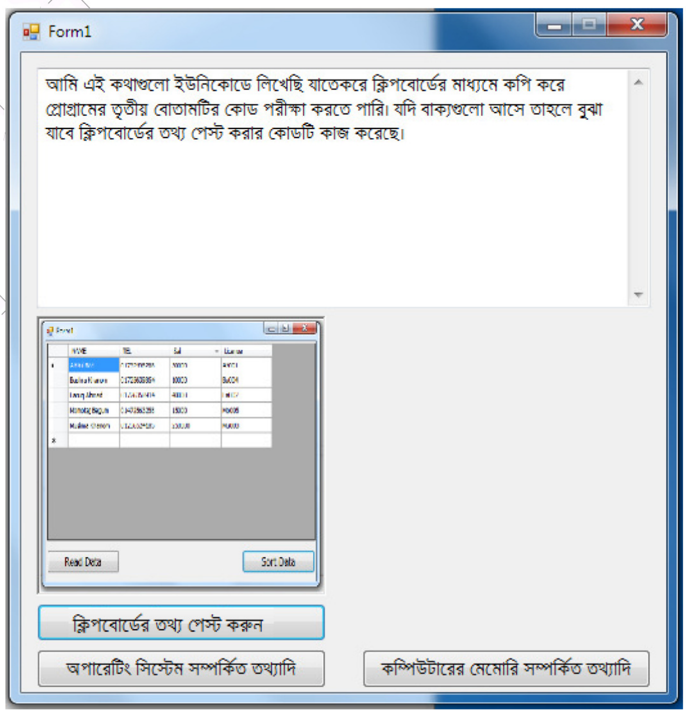
১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

(ঝ) My.Computer.Clipboard এই ফাংশনের মাধ্যমে আমরা ক্লিপবোর্ড সম্পর্কে তথ্য জানা ও একে নিয়ন্ত্রণ করতে পারি। উদাহরণ হিসাবে নিচের কোডগুলো দেখুন।

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    If My.Computer.Clipboard.ContainsText Then 'check if the clipboard contains text
        TextBox1.Text = ""
        TextBox1.Text = My.Computer.Clipboard.GetText 'paste into TextBox
    End If
    If My.Computer.Clipboard.ContainsImage Then 'check if the clipboard contains Image
        PictureBox1.Image = My.Computer.Clipboard.GetImage 'paste into PictureBox
    End If
End Sub
```

উক্ত সাবের ফাংশনগুলো কিভাবে কাজ করে তা দেখতে চাইলে আগের পৃষ্ঠায় প্রদর্শিত ফর্মে তৃতীয় একটি বোতাম যুক্ত করুন। আরো যুক্ত করুন একটি পিকচারবক্স। বোতামের টেক্সট যা ইচ্ছে দিতে পারেন পিকাচর বাক্সের SizeMode প্রোপার্টি হবে StretchImage । নতুন বোতামে ডবল ক্লিক করে উক্ত সাবটি লিখুন। এবার প্রোগ্রাম রান করুন। ক্লিপবোর্ডে ছবি বা লেখা কপি করুন যে কোনো ফাইল খুলে। ক্লিক করুন তৃতীয় বোতামে। ছবি থাকলে পিকচার বাক্সে আসবে, লেখা থাকলে টেক্সটবাক্সে আসবে। কিছু না থাকলে কিছুই আসবে না। নিচের ছবিতে প্রোগ্রাম রানকালে ক্লিপবোর্ড থেকে টেক্সট ও চিত্র আনার দৃশ্য দেখাচ্ছে।



(ঞ) My.Computer.Clock এই ক্লাসে ৩টি প্রপার্টি আছে যা আমরা কাজে লাগাতে পারি। এ প্রপার্টিগুলো হলো:

(১) My.Computer.Clock.GmtTime - উদাহরণ: MsgBox("Current GMT time: " & My.Computer.Clock.GmtTime) 'যে কম্পিউটারে এই কোডটি রান করবে, সেটির জিমএমটি [Greenwich Mean Time] সময় দেবে।

(২) **My.Computer.Clock.LocalTime** -এ প্রপার্টি থেকে স্থানীয় সময় পাওয়া যায়। যেমন:
MsgBox("Current local time: " & My.Computer.Clock.LocalTime)

(৩) **My.Computer.Clock.TickCount** -এ প্রপার্টি থেকে কম্পিউটারের ভেতরের টাইমারের মিলিসেকেন্ড গণনা পাওয়া যায়। যেমন:
Label1.Text = My.Computer.Clock.TickCount 'এ কোড রান করার সময় কম্পিউটারের অভ্যন্তরীণ টাইমারের মিলিসেকেন্ড সংখ্যা লেবেলে প্রদর্শিত হবে।

(ট) **My.Computer.Keyboard** -এই ক্লাসে বেশ ক'টি প্রপার্টি ও মেথড আছে যা আমরা কাজে লাগাতে পারি। কম্পিউটারের কিবোর্ড সম্পর্কিত কয়েকটি প্রপার্টি হলো:

(১) **My.Computer.Keyboard.AltKeyDown** -অলটার কি প্রেস করা আছে কি না সে সম্পর্কে 'সত্য' বা 'মিথ্যা' বুলিয়ান ফলাফল দেবে। যেমন:
If My.Computer.Keyboard.AltKeyDown Then MsgBox("ALT key down")

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

(২) **My.Computer.Keyboard.CapsLock** -ক্যাপ্স লক প্রেস করা আছে কি না সে সম্পর্কে 'সত্য' বা 'মিথ্যা' বুলিয়ান ফলাফল দেবে। যেমন:

```
If My.Computer.Keyboard.CapsLock Then MsgBox("Caps Lock is ON")
```

(৩) **My.Computer.Keyboard.CtrlKeyDown** -কন্ট্রোল কি প্রেস করা আছে কি না সে সম্পর্কে 'সত্য' বা 'মিথ্যা' বুলিয়ান ফলাফল দেবে। যেমন:

```
If My.Computer.Keyboard.CtrlKeyDown Then MsgBox("Control key is down")
```

এভাবে আরো আছে, যাদের ফলাফল হচ্ছে বর্ণিত কি-টি চাপানো আছে কি না:

(৪) My.Computer.Keyboard.**NumLock**

(৫) My.Computer.Keyboard.**CtrlScrollLock**

(৬) এবং My.Computer.Keyboard.**ShiftKeyDown**

(ঠ) **My.Computer.Mouse** -এটির দ্বারা কম্পিউটারের মাউস সম্পর্কে জানা ও সেট করার দু'টি প্রোপার্টি জানা যায়। যেমন:

(১) **My.Computer.Mouse.WheelExists** -মাউসের মধ্যে স্ক্রল চাকা আছে কি না? থাকলে '**True**' না থাকলে '**False**'।

(২) **My.Computer.Mouse.WheelScrollLines** -মাউসের স্ক্রল চাকা প্রতি চক্করে কতটুকু ঘুরবে সে তথ্য দেবে। উদাহরণ হিসেবে নিচের কোড ব্লক ব্যবহার করে দেখুন। একটি বোতামের ইভেন্ট কোডে এটি লিখে রান করুন।

```
If My.Computer.Mouse.WheelExists Then
   Dim lines As Integer = My.Computer.Mouse.WheelScrollLines
   If lines > 0 Then
      MsgBox("Application scrolls " &
         lines & " line(s) for each wheel turn.")
   Else
      MsgBox("Application scrolls " &
         (-lines) & " page(s) for each wheel turn.")
   End If
Else
   MsgBox("Mouse has no scroll wheel.")
End If
```

(ঠ) **My.Computer.Network -** যে কম্পিউটারে কোড রান করবে তাতে নেটওয়ার্ক আছে কি না, থাকলে নির্দিষ্ট ইউআরএল ঠিকানা থেকে ফাইল ডাইনলোড করা ও ফাইল আপলোড করা ইত্যাদি কাজ আঞ্জাম দেওয়ার সুযোগ করে দিয়েছে এই মেথডটি। নিচে আমরা তিনটি ফাংশন উদাহরণসহ তুলে ধরেছি।

(১) My.Computer.Network.IsAvailable – এটা দ্বারা আমরা জানতে পারি বর্তমান সময়ে নেটওয়ার্ক আছে কি না? যেমন:

```
If My.Computer.Network.IsAvailable = True Then
   MsgBox("Computer is connected.")
Else
   MsgBox("Computer is not connected.")
End If
```

১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

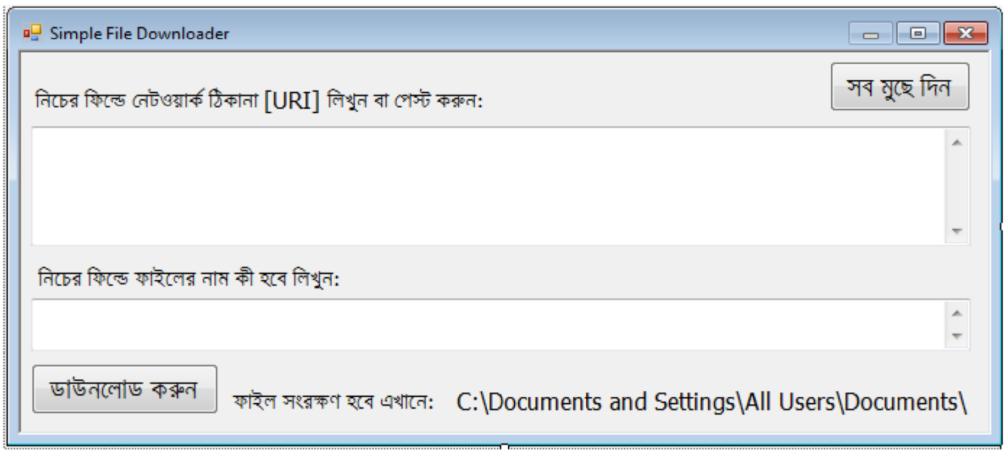
(২) **My.Computer.Network.DownlaodFile** -এই মেথডের কাজ হলো নির্দিষ্ট নেটওয়ার্ক ঠিকানার ফাইল ডাউনলোড করে নির্দিষ্ট জায়গায় সংরক্ষণ করা। যেমন:

```
My.Computer.Network.DownloadFile("http://sufiaekram.blogspot.com/p/junaid-baghdadira.html", "C:\Documents and Settings\All Users\Documents\Junaid Bagdadi (RA).html", "", "", True, 500, True)
```

উক্ত কোডের প্রথম আর্গুমেন্ট হলো ফাইলের ঠিকানা। দ্বিতীয়টি হলো কম্পিউটারের কোথায় তা ডাউনলোড হয়ে সংরক্ষিত হবে তার পূর্ণ প্যাথ। তৃতীয় ও চতুর্থ আর্গুমেন্ট হলো, যথাক্রমে ইউজার নেইম ও পাসওয়ার্ড [এ ক্ষেত্রে কিছুই নয় দেখানোর জন্য "" দেওয়া হয়েছে]। ৫ম আর্গুমেন্ট দ্বারা আমরা ডাউনলোড প্রগ্রেস বার দেখানোর নির্দেশ দিয়েছি। ৬ষ্ট আর্গুমেন্ট হলো টাইম-আউট লিমিট -এখানে ৫০০ মিলিসেকেন্ড এবং ৭ম আর্গুমেন্ট দ্বারা বলা হচ্ছে কম্পিউটারে এই একই ফাইল একই লকেশনে থাকলে তা অভাররাইট করা যাবে [এখানে False লিখলে অভাররাইট করা যাবে না- এই নির্দেশ দেওয়া হবে]।

আসুন, একটি ব্যবহারযোগ্য এ্যাপ্লিকেশন তৈরি করি, কেমন? আমাদের উদ্দেশ্য হলো, যে কোনো ওয়েভসাইট থেকে ডাউনলোড করা যায় এমন কোনো ফাইল ডাউনলোড করে আমাদের কম্পিউটারে সংরক্ষণ করতে। সুতরাং নিচের পদক্ষেপগুলো অনুসরণ করে আমরা প্রজেক্টটি তৈরি করবো।

ক. একটি নতুন প্রজেক্ট তৈরি করুন। এর নাকরণ হবে: SimpleFileDownloader।
খ. নিচের ছবি মুতাবিক ফর্ম ডিজাইন করুন। এতে থাকবে: ৪টি লেবেল, দু'টি টেক্সট্ বাক্স ও দু'টি বোতাম। ফর্মের FormBorderStyle হবে Fixed3D, টেক্সট্ প্রপার্টি হবে Simple File Downloader, এবং MaximizeBox প্রপার্টি হবে False।
গ. উপরের লেবেলের টেক্সট্ প্রপার্টিতে লিখুন: নিচের ফিল্ডে নেটওয়ার্ক ঠিকানা [URI] লিখুন বা পেস্ট করুন:। মাঝেরটিতে লিখুন: নিচের ফিল্ডে ফাইলের নাম কী হবে লিখুন:| নিচের বায়েরটিতে লিখুন: ফাইল সংরক্ষণ হবে এখানে:। সবশেষে নিচের ডানেরটিতে লিখুন: C:\Documents and Settings\All Users\Documents\।
নোট: আপনি চাইলে এসব লেবেলে ভিন্নভাবেও লিখতে পারেন- কিংবা ইংরেজিতেও লিখতে পারেন। আমি ইউনিকোডে বাংলায় লিখেছি। তবে শেষেরটিতে একটি প্যাথ নাম লিখেছি। ফাইল সংরক্ষণ করতে অন্য কোনো ফল্ডারও ব্যবহার করতে পারেন। সেক্ষেত্রে ফল্ডারের পূর্ণ প্যাথ নাম লিখতে হবে শেষোক্ত ফিল্ডে।
ঘ. উপরের ডানের বোতামের টেক্সট্ প্রপার্টিতে লিখুন: নতুন ফাইল ডাউনলোড করুন। দ্বিতীয়টির টেক্সট্ প্রপার্টিতে লিখুন: সব মুছে দিন| ব্যস! নিচের ছাবিতে দেখুন আমাদের ইউজার ইন্টারফেইস ডিজাইনটি।



১২তম পরিচ্ছেদ: *Accessing Computer Resources*

কম্পিউটারের রিসোর্স সংগ্রহ ও ব্যবহার : *Accessing Computer Resources*

এবার বোতাম দু'টির ক্লিক ইভেন্টে আমরা কোড লিখবো। প্রথমে ডবল ক্লিক করুন 'সব মুছে দিন' বোতামে। ক্লিক ইভেন্টে নিচের কোডটি লিখুন।

```
'clear text for new entry
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
      TextBox1.Text = ""
      TextBox2.Text = ""
End Sub
```

উপরের কোডগুলো সহজেই অনুমেয়। আমরা উভয় টেক্সট বাক্স থেকে লেখা মুছে দিয়েছি। এবার ডবল ক্লিক করুন 'নতুন ফাইল ডাউনলোড করুন' বোতামে। ক্লিক ইভেন্টে লিখুন:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      Dim FIleURI As String = TextBox1.Text
      Dim FileSaveLoc As String = Label4.Text & TextBox2.Text
      If FIleURI = "" Or TextBox2.Text = "" Then
         MsgBox("Please Enter URI address or File Name. One or Both are MISSING!!",
MsgBoxStyle.Exclamation)
         Exit Sub
      End If
      On Error GoTo ThisLine
      My.Computer.Network.DownloadFile(FIleURI, FileSaveLoc, "", "", True, 500, True)
      MsgBox("File has been downloaded at: " & FileSaveLoc)
      Exit Sub
ThisLine:
      MsgBox("Sorry! An error occured. Please Try again. Make sure to enter correct texts
into fields.", MsgBoxStyle.Information)
   End Sub
```

দ্বিতীয় এ কোড ব্লকটিই মূল কোড। এগুলো লাইন বাই লাইন বুঝিয়ে বলছি। প্রথমত আমরা টেক্সটবাক্স-১ এ যে টেক্সটটুকু আছে তা একটি স্ট্রিং ভেরিয়েবলে সংরক্ষণ করেছি। অনুরূপ আরেকটি ভেরিয়েবলে সংরক্ষণ করেছি লেবেল৪ ও টেক্সট২ এর লেখাটুকু। লক্ষ করুন আমরা নির্দেশ দিচ্ছি লেবেলের টেক্সটের সাথে টেক্সটবক্স২ এর টেক্সট যোগ করতে। উভয়টি মিলে ফাইল সংরক্ষণের পূর্ণাঙ্গ প্যাথ সংরক্ষিত হবে FileSaveLoc ভেরিয়েবলে। ব্যবহারকারী কোনো একটি বা উভয় টেক্সট বাক্সে কিছু না লিখলে এ্যারর আসবে। তাই আমরা If - End If ব্লকে নিশ্চিত করছি ব্যবহারকারী উভয় ফিল্ডে টেক্সট লিখেছেন। না লিখলে একটি ম্যাসেজ আসবে। এরপর কোড রান করা এখানেই শেষ হবে।

On Error GoTo ThisLine -একটি পুরাতন কোডিং প্রাকটিস যা এখনো জনপ্রিয়। এখানে যা বলা হচ্ছে তাহলো: পরের কোড লাইনটিতে কোনো এ্যারর হলে ThisLine: -নামক কোডে জাম্প দাও। সেখায়ে যেয়ে একটি মেসেজ বাক্সে ভুলের রিপোর্ট প্রদর্শিত হবে। এরপর কোড রানিং শেষ হবে। এভাবে করার কারণ হলো, যাতে করে এ্যাররের ফলে পুরো এ্যাপলিকেশন বন্ধ বা ক্রাশ না করে।

যাক, যে কোড লাইনটি আসল কাজ তথা ফাইল ডাউনলোড করে তাহলো:

```
My.Computer.Network.DownloadFile(FIleURI, FileSaveLoc, "", "", True, 500, True)
```

আমরা ইতোমধ্যে জেনেছি এই মেথডের ৭টি আর্গুমেন্ট মানে কি। সুতরাং এখানে আর পুনরাবৃত্তি করছি না। তবে এটুকু বলছি যে, ব্যবহারকারী কর্তৃক লিখিত বা ওয়েবসাইট থেকে পেস্ট করা ফাইল হলো FIleURI। অনুরূপ ব্যবহারকারী কর্তৃক নির্দেশিত ফাইল নামসহ সংরক্ষণের পূর্ণ প্যাথ হলো FileSaveLoc।

১৩তম পরিচ্ছেদ: *Software Development Cycle*

সফ্‌টওয়্যার তৈরির প্রক্রিয়াচক্র : *Software Development Cycle*



ব্যবহারযোগ্য বা কমার্শিয়্যাল সফ্‌টওয়্যার তৈরি করতে প্রোগ্রামাররা একটি সুচিন্তিত প্রণালীবদ্ধ পদ্ধতি অনুসরণ করে থাকেন। এতে আছে বেশ ক'টি পদক্ষেপ। সঠিকভাবে এগুলো অনুসরণ করলে কাঙ্ক্ষিত ফলাফল পাওয়ার নিশ্চয়তা থাকে। আমরা এরূপ ৬টি পদক্ষেপ সনাক্ত করেছি যাকে, বলা যায়: সফ্‌টওয়্যার তৈরির প্রক্রিয়াচক্র (বায়ের চিত্রটি দেখুন)। ইংরেজিতে বলতে পারি:*Software Development Cycle*।

পরের দীর্ঘ পরিচ্ছেদে আমরা বেশ ক'টি ব্যবহারযোগ্য সফ্‌টওয়্যার ডেভেলাপ করবো। শিক্ষার্থীরা বায়ের চিত্রে প্রদর্শিত সব ক'টি পদক্ষেপ নিয়মতান্ত্রিণভাবে অনুসরণ করবেন- এটাই কাম্য। প্রত্যেক সফ্‌টওয়্যার তৈরির প্রক্রিয়া খুব স্পষ্ট ভাষায় ধারাবাহিকভাবে বর্ণিত হবে অবশ্যই। প্রথম প্রজেক্ট শুরুর আগে আসুন, আমরা এ ৬টি পদক্ষেপের ওপর আরো কিছু ব্যাখ্যা তুলে ধরি।

ক. প্রয়োজন সনাক্তকরণ - Identify Requirement

প্রত্যেক সফ্‌টওয়্যার সৃষ্টি করা হয় কোনো এক বা একাধিক প্রয়োজন মেটানোর উদ্দেশ্যে। যেমন: মাইক্রসফটের অফিস ২০০৭ সফ্‌টওয়্যারের মধ্যে মোট ৯টি ভিন্ন প্রোগ্রাম আছে। এগুলো হলো: Access, Excel, Groove, InfoPath, OneNote, Outlook, PowerPoint, Publisher and Word। এদের প্রত্যেকটির কাজ ভিন্ন। অফিস ডেভেলাপমেন্ট প্রোগ্রামাররা কম্পিটারের সাহায্যে বিভিন্ন কাজ কিভাবে আঞ্জাম দেওয়া যায় তা সনাক্ত করেছেন। কোন্ প্রয়োজন কোন্ প্রোগ্রাম মেটাবে তা সনাক্তকরণের নামই হলো Identify Requirement। আমাদের ১ম ব্যবহারযোগ্য প্রোগ্রামের নাম: 'বাংলাদেশ শহর থেকে শহর দূরত্ব' [BD City to City Distance]। এটি সৃষ্টির পূর্বে প্রয়োজন সনাক্তকরণ করতে যেয়ে আমরা দেশের ৬৪টি জিলা শহরের মধ্যে দূরত্ব কিলোমিটার ও মাইলের হিসাবে জানার স্পৃহার কথা ভেবেছি। সুতরাং এ প্রয়োজন মেটানোর জন্যই আমরা প্রোগ্রাম ডেভেলাপমেন্টে অগ্রসর হয়েছি।

খ. বিশ্লেষণ - Analysis

প্রয়োজন সনাক্তকরণের পর বুঝতে হবে কিভাবে তা মেটানো যায়। এজন্য প্রয়োজন সঠিক ব্যাখ্যার। আর এ পদক্ষেপের নামই হলো বিশ্লেষণ। আমাদের ১ম প্রোগ্রামের বিশ্লেষণ হলো: ব্যবহারকারী যে কোনো দু'টি শহরের নাম চয়েজ করে জেনে নেবেন এদের মধ্যে রোডের দূরত্ব কী। দূরত্বটি হবে কিলোমিটার ও মাইলের এককে।

গ. আকল্প - Design

উপরের উভয় পদক্ষেপগুলো থেকে সঠিক জবাব পাওয়ার পরই আসল ডেভেলাপমেন্ট প্রক্রিয়া শুরু হবে। প্রথমে অবশ্যই ইউজার ইন্টারফেইস তৈরি করতে হবে। আর এ ইউআই সৃষ্টির নামই হলো প্রোগ্রাম ডিজাইন পদক্ষেপ বা আকল্প। আমাদের ১ম প্রোগ্রামের ডিজাইনটি দেখলেই বুঝতে পারবেন এ স্টেজে আসলে কী করতে হয়। একটু পরই তা দেখতে পারবেন।

১৩তম পরিচ্ছেদ: ***Software Development Cycle***

সফ্টওয়্যার তৈরির প্রক্রিয়াচক্র : *Software Development Cycle*

ঘ. সংক্ষেতলিপি - Coding

ডিজাইন স্টেজ শেষ করার পর প্রোগ্রামিংয়ের আসল কাজে হাত দিতে হবে, অর্থাৎ কোড লিখা। ব্যবহারকারী প্রোগ্রামের সাথে মিথস্ক্রিয়া করে কাঙ্ক্ষিত ফলাফল পাবে- এটাই তো সফ্টওয়্যার তৈরির মূল লক্ষ্য। অন্যকথায়, কাঙ্ক্ষিত ফলাফল দৃশ্যত হবে, কিংবা প্রিন্টারের মাধ্যমে কাগজে অঙ্কিত হবে বা ইন্টারনেটের মাধ্যমে অন্যত্র আপলোড করা হবে ইত্যাদি। তবে কাঙ্ক্ষিত ফলাফল অর্জনে প্রয়োজন হয় কিছু ক্রিয়া বা এ্যাকশনের। এসব ক্রিয়াকে সামগ্রিকভাবে বলা যায় **events** বা ঘটনাপ্রবাহ। বাস্তবে পুরো কম্পিউটার মূলত ইভেন্ট দ্বারা নিয়ন্ত্রিত হয়। প্রোগ্রামার হিসাবে আমরা এসব ইভেন্ট বা ঘটনাকে ক্যাচ্ করে নিজের ইচ্ছেমতো পরিচালনা করি। এটার নামই কোডিং। একটি উদাহরণ দেখুন। মনে করুন আপনার ফর্মে একটি বোতাম ও একটি লেবেল আছে। আপনার উদ্দেশ্য বোতামের ক্লিক নামক ইভেন্টকে কাজে লাগিয়ে ঐ লেবেলে কিছু লেখা প্রদর্শন করবেন। এ লক্ষ্যর্জনে ক্লিক ইভেন্টে এক লাইনের কোড লিখতে পারেন। যেমন:-

```
label1.text ="আমি বোতাম-১ এর ক্লিক ইভেন্ট ধরে এই কথাগুলো লিখেছি!"
```

প্রোগ্রাম চলাকালে বোতামে ক্লিক করলেই ইভেন্ট সক্রিয় হবে। আর যেহেতু আপনি একে ধরে বলছেন লেবেলে টেক্সট্ প্রদর্শনের জন্য, তাই এ কাজ করা ইভেন্টের জন্য বাধ্যতামূলক হয়ে যাবে।
অনুরূপ মাউস, কিবোর্ড, প্রিন্টার ইত্যাদি সবকিছু পরিচালনার জন্য একাধিক ইভেন্ট আছে। ভিজুয়্যাল বেসিক.নেট এসব ইভেন্ট ধরে কাজে লাগানের সুযোগ সৃষ্টি করে দিয়েছে। বিশেষ নিয়ম-পদ্ধতি অনুসরণ করে আপনি কোড লিখলে ইভেন্ট কাজে লাগিয়ে কাঙ্ক্ষিত ফলাফল পেতে পারেন।

ঙ. পরীক্ষণ - Testing

উপরে ব্যাখ্যাত কোডিং স্টেজটিই হলো সফ্টওয়্যার তৈরির মূল অংশ। এর পরের দু'টির প্রথমটি পরীক্ষণ স্টেজ। এ ব্যাপারে বেশি বলার প্রয়োজন নেই। কারণ, যে কোনো নতুন সৃষ্টি সঠিকভাবে ক্রিয়া করছে কি না তা বার বার পরীক্ষা না করে উপায় নেই। এমনকি কোডিং স্টেজেও বার বার প্রোগ্রাম রান (পরিচালনা) করে দেখতে হবে বিভিন্ন কোড ব্লক সঠিকভাবে কাজ করছে কি না? পুরো প্রোগ্রাম শেষ করার পরও বার বার টেস্ট করতে হয় বিভিন্ন অবস্থাভেদে। আপনি তো চাইবেন না, প্রোগ্রামের মধ্যে কেনো বাগ (খারাপ কোডিং) কিংবা এক্সসেপশন (ক্রাশ হওয়ার সম্ভাবনা) থাকুক! সুতরাং যে কোনো কন্ডিশনে যাতে আপনার প্রোগ্রামটি গ্রাহকের হাতে যেয়ে সঠিকভাবে চলে তা নিশ্চিত করতে যেয়ে আপনাকে বার বার প্রোগ্রাম পরিচালনা করে টেস্ট করতে হবে। এরূপ টেস্টিংকে বলে ডিবাগিং [debugging]।

চ. বিস্তারণ - Deployment

সর্বশেষ এ স্তরে পৌঁছুনোর পরই বুঝতে হবে সফ্টওয়্যারটি সৃষ্টিতে সফলতা লাভ হয়েছে। কারণ, এখন একটি মাত্র কাজ বাকী রয়েছে। সেটা হলো, গ্রাহকের নিকট পৌঁছুনো। এ কাজের জন্য ভিজুয়্যাল স্টুডিও খুব সহজ একটি উপায় সৃষ্টি করে দিয়েছে। মাত্র কয়েকটি পদক্ষেপে আপনাার সৃষ্ট সফ্টওয়্যারটি সিডি, পেন ড্রাইভ বা নেটওয়ার্কের মাধ্যমে বিস্তারণ করতে ইন্সটলার ফাইল বানিয়ে দেবে। আপনার সফ্টওয়্যারের জন্য প্রয়োজনীয় অন্যান্য ফাইলও প্যাকেজ করে দেবে যাতে করে ইন্সটলার ফাইল সহজে এগুলো গ্রাহকের কম্পিউটারে নির্দিষ্ট স্থানে সংরক্ষণ করে। এসব অতিরিক্ত ফাইলকে বলে **dependency files**। আপনার প্রোগ্রামে থাকতে পারে এক বা একাধিক টেক্সট্ ফাইল। প্রোগ্রাম চলাকালে এসব ফাইল থেকে তথ্য লোড করা জরুরী হতে পারে। সুতরাং এসব ফাইলও নির্দিষ্ট ফল্ডারে সংরক্ষিত থাকা প্রয়োজন।

যাক, এবার আমরা ১ম 'গ্রাহক ব্যবহারযোগ্য' একটি সফ্টওয়্যার তৈরির কাজে অগ্রসর হবো। শিক্ষার্থীদের ধৈর্যধারণ করা একান্ত কাম্য- কারণ, ১ম এই সফ্টওয়্যারের জন্য কয়েক শত লাইন কোডিং করতে হবে! তবে হতাশ হবেন না, অধিকাংশ ক্ষেত্রে কোড হবে কপি-পেস্ট! অবশ্য বেশ কয়েক পৃষ্ঠাব্যাপী প্রোগ্রামের ব্যাখ্য এসেছে- ধৈর্য ও মনোযোগসহ এ পৃষ্ঠাগুলো পাঠ করতে হবে। সেসাথে আইডিই খুলে নির্দেশিত পদক্ষেপ অনুসরণ করে সফ্টওয়্যার বানিয়েও যেতে হবে।

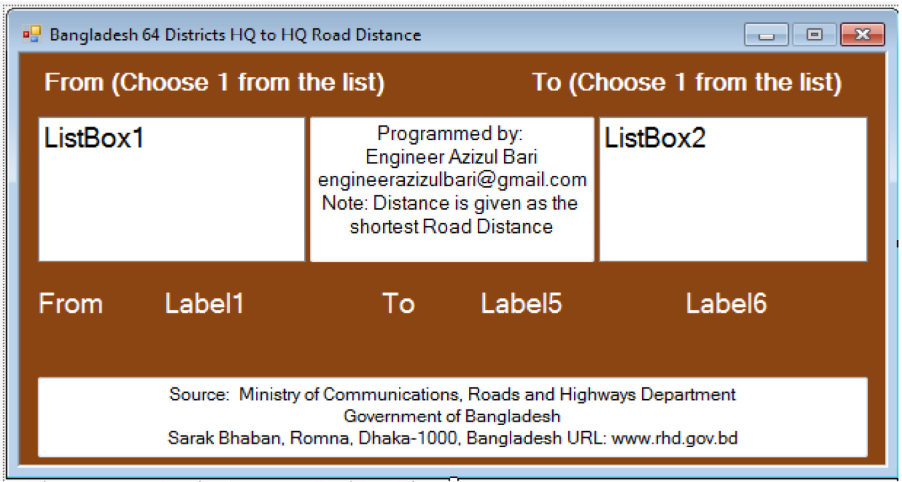
পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

Application 1: BD City to City Distance - বাংলাদেশ শহর থেকে শহর দূরত্ব

উদ্দেশ্য: আমাদের উদ্দেশ্য হলো, ব্যবহারকারী বাংলাদেশের ৬৪টি জিলা শহরের রোড-দূরত্ব সহজে জানতে পারবেন। দূরত্বে প্রদর্শিত হবে কিলোমিটার ও মাইল এককে।

ডিজাইন: প্রথমেই আমরা একটি আকর্ষণীয় ইউজার ইন্টারফেইস ডিজাইন করবো। বায়ের চিত্রটি দেখুন ও পদক্ষেপগুলো অনুসরণ করুন।

১. একটি নতুন প্রজেক্ট তৈরি করুন। নামকরণ করুন: "BD City to City Distance"।
২. বায়ের চিত্রের মতো ফর্মটি ডিজাইন করুন।



ফর্মের বস্তুসমূহ: ৭টি লেবেল, ২টি লিস্ট-বাক্স এবং (চাইলে) দু'টি টেক্স-বাক্স।

ফর্মটি যাতে একই মাপের থাকে সেজন্য এর FormBorderStyle = Fixed3D, MaximizeBox = False and AutoScaleMode = None হবে। এর Text প্রপার্টি হবে Bangladesh 64 Districts HQ to HQ Road Distance। আপনি চাইলে অন্য নামও দিতে পারেন। রং আমি একটা চয়েজ করেছি। আপনি যা চান সেটাই দিতে পারেন। সবশেষে দু'টি ফিল্ডে কিছু তথ্য দিয়েছি। নিচেরটি জরুরী বলে মনে করি। আর উপরেরটি না দিলেই হবে। অবশ্য থাকলে তো আপনার নিজের নামই থাকবে, আমার পরিবর্তে! বিভিন্ন লেবেলে যা লিখা আছে তা-ই (টেক্স্ট) প্রোপার্টিতে লিখুন। ব্যস! ডিজাইন হয়ে গেলো। এবার আসল কাজ- কোডিং!

প্রোগ্রামের তথ্যাদি: আমাদের এই বিশেষ প্রোগ্রামটি কিছু তথ্যের উপর নির্ভরশীল। আগে থেকেই আমাদেরকে ৬৪টি জিলার নামসহ একটা থেকে আরেকটির দূরত্ব (কিলোমিটার এককে) জানতে হবে। এজন্য আমরা যোগাযোগ মন্ত্রণালয়ের ওয়েভসাইটটি কাজে লাগিয়েছি। সেখানে দেওয়া আছে একটি টেবিল। এতে জিলাগুলোর নাম ও একটা থেকে আরেকটির দূরত্ব মেট্রিক্স পদ্ধতিতে লিখিত আছে। এই তথ্যগুলোর উপর নির্ভর করেই আমরা পুরো এ্যাপলিকেশনটি তৈরি করেছি। একজন ব্যবহারকারী প্রশ্ন করতে পারেন: দূরত্ব আপনি পেলেন কোথায়? এগুলো কি সঠিক? আমাদেরকে জবাব দিতে হবে না। বাংলাদেশের তথ্য মন্ত্রণালয় থেকেই যে তথ্যগুলো পেয়েছি তা-তো আমরা নিচের টেক্স বাক্সে ঘোষণা দিয়েছি! যাক এবার আসুন কোডগুলোর ওপর আলোচনা শুরু করি। আপনি শহরের নাম ও দূরত্ব সংখ্যা টাইপ করতে সতর্কতা অবলম্বন করবেন- তথ্যে ভুল হলে হয় প্রোগ্রাম কাজ করবে না, কিংবা ভুল ফলাফল পাবেন। আমরা এর কোনোটাই চাই না।

২. কোডিং: ফর্মে ডবল-ক্লিক করে কোড জানালা খুলুন। প্রথমেই আমাদেরকে কয়েকটি গ্লোব্যাল ভেরিয়েবল ডিক্লায়ার করতে হবে। নিচের (পরের পৃষ্ঠায় দেওয়া) কোড লাইন মুতাবিক এগুলো ডিক্লায়ার করুন।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Public Class Form1
    'put distance of citys in array variables as matrices - i.e. 0 km to 0 index as distance to other city: Dhaka 0 to Bhola 205 (both 0 index)
    Dim Dhaka() As String = {"0", "178", "316", "247", "169", "205", "197", "109", "115", "242", "215", "96", "391", "338", "101", "149", "268", "37", "127", "163", "179", "164", "182", "178", "249", "259", "180", "117", "348", "183", "137", "343", "90", "150", "63", "240", "203", "27", "122", "247", "130", "17", "51", "210", "302", "158", "359", "158", "216", "443", "204", "185", "118", "256", "293", "304", "240", "101", "188", "134", "296", "241", "92", "407"}
```

লক্ষ করুন: Dhaka() একটি স্ট্রিং এ্যারে ভেরিয়েবল যার মধ্যে আছে ৬৪টি ইলেমেন্ট (ইনডেক্স ০ থেকে ৬৩)। প্রোগ্রাম চলাকালে এই ভেরিয়েবল [এবং শহরগুলোর নামে যাবতীয় এ্যারে ভেরিয়েবল] মেমোরিতে সংরক্ষিত থাকবে এবং এগুলো প্রোগ্রামের যে কোনো জায়গা থেকে ব্যবহারও করা যাবে- যা আমাদের ইচ্ছে। এবার বাকী শহরগুলোর এ্যারে ভেরিবেল ডিক্লায়ার করুন:

```
    Dim Bagerhat() As String = {"178", "0", "437", "160", "119", "155", "328", "283", "182", "363", "175", "269", "512", "467", "140", "270", "397", "215", "52", "336", "357", "93", "100", "139", "378", "381", "34", "290", "477", "184", "209", "472", "110", "136", "200", "201", "376", "179", "300", "357", "107", "186", "225", "260", "352", "336", "488", "244", "234", "572", "151", "66", "167", "306", "415", "433", "94", "134", "366", "311", "470", "414", "270", "536"}
    Dim Bandarban() As String = {"316", "437", "0", "469", "391", "427", "510", "301", "271", "85", "526", "225", "121", "651", "413", "167", "581", "350", "386", "376", "492", "476", "404", "490", "563", "133", "439", "379", "661", "499", "229", "656", "327", "461", "379", "552", "416", "329", "436", "561", "442", "314", "339", "524", "616", "444", "672", "208", "530", "757", "426", "438", "434", "569", "74", "618", "499", "317", "501", "447", "509", "454", "406", "720"}
    Dim Barguna() As String = {"247", "160", "469", "0", "84", "120", "445", "352", "214", "396", "323", "281", "544", "584", "209", "303", "513", "284", "196", "406", "426", "209", "95", "286", "495", "413", "150", "360", "594", "301", "241", "589", "143", "258", "270", "348", "446", "248", "370", "473", "227", "255", "294", "377", "469", "406", "604", "276", "350", "689", "44", "96", "236", "423", "447", "550", "206", "166", "435", "381", "539", "483", "340", "652"}
    Dim Barisal() As String = {"169", "119", "391", "84", "0", "36", "366", "273", "135", "317", "244", "202", "465", "505", "131", "224", "435", "206", "117", "327", "348", "168", "19", "207", "416", "334", "109", "281", "515", "222", "162", "510", "64", "179", "191", "270", "367", "169", "291", "395", "148", "177", "215", "298", "390", "327", "526", "197", "272", "610", "41", "53", "158", "344", "368", "471", "164", "87", "357", "302", "460", "405", "261", "574"}
    Dim Bhola() As String = {"205", "155", "427", "120", "36", "0", "402", "309", "171", "353", "280", "238", "501", "541", "167", "260", "471", "242", "153", "363", "384", "204", "55", "243", "452", "370", "145", "317", "551", "258", "198", "546", "100", "215", "227", "306", "403", "205", "327", "431", "184", "213", "251", "334", "426", "363", "562", "233", "308", "646", "77", "89", "194", "380", "404", "507", "200", "123", "393", "338", "496", "441", "297", "610"}
    Dim Bogra() As String = {"197", "328", "510", "445", "366", "402", "0", "274", "309", "437", "192", "291", "585", "141", "239", "344", "71", "169", "324", "328", "172", "235", "379", "190", "52", "454", "295", "201", "151", "144", "331", "146", "287", "217", "188", "182", "368", "221", "178", "50", "269", "211", "217", "68", "160", "217", "162", "353", "127", "246", "401", "350", "208", "114", "488", "107", "307", "298", "188", "72", "462", "406", "105", "210"}
    Dim Brahmanbaria() As String = {"109", "283", "301", "352", "273", "309", "274", "0", "143", "227", "320", "81", "375", "416", "206", "134", "345", "114", "231", "76", "203", "269", "287", "283", "327", "244", "285", "79", "425", "268", "164", "420", "195", "254", "148", "326", "116", "123", "147", "325", "235", "107", "67", "288", "380", "144", "436", "143", "294", "521", "308", "290", "204", "334", "278", "382", "344", "205", "204", "211", "209", "153", "170", "484"}
    Dim Chandpur() As String = {"115", "182", "271", "214", "135", "171", "309", "143", "0", "198", "257", "67", "346", "450", "144", "105", "380", "149", "130", "218", "291", "195", "149", "220", "361", "215", "184", "221", "460", "235", "43", "455", "71", "192", "177", "283", "258", "128", "234", "359", "162", "113", "137", "311", "403", "270", "471", "78", "285", "555", "170", "182", "171", "357", "249", "416", "243", "62", "300", "246", "351", "295", "204", "519"}
    Dim Chittagong() As String = {"242", "363", "85", "396", "317", "353", "437", "227", "198", "0", "453", "151", "160", "578", "339", "93", "507", "276", "312", "302", "418", "402", "330", "416", "489", "117", "366", "305", "588", "425", "155", "583", "253", "387", "305", "478", "342", "256", "362", "487", "368", "240", "265", "450", "542", "370", "598", "134", "456", "683", "352", "364", "360", "496", "78", "544", "425", "244", "427", "373", "436", "380", "332", "646"}
    Dim Chuadanga() As String = {"215", "175", "526", "323", "244", "280", "192", "320", "257", "453", "0", "306", "601", "331", "118", "360", "261", "237", "171", "373", "276", "82", "231", "37", "243", "470", "142", "305", "341", "49", "284", "336", "186", "65", "173", "26", "413", "216", "282", "221", "116", "223", "262", "124", "217", "321", "352", "319", "98", "437", "279", "198", "108", "170", "504", "297", "154", "209", "291", "176", "507", "451", "209", "400"}
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Dim Comilla() As String = {"96", "269", "225", "281", "202", "238", "291", "81", "67", "151", "306", "0", "300", "432",
"193", "58", "361", "130", "218", "156", "272", "256", "215", "270", "343", "169", "272", "159", "441", "279", "89", "436",
"138", "241", "159", "332", "196", "109", "216", "341", "222", "94", "119", "304", "396", "224", "452", "67", "310", "537",
"237", "249", "214", "350", "203", "398", "331", "128", "281", "227", "290", "234", "186", "500"}
Dim coxsBazar() As String = {"391", "512", "121", "544", "465", "501", "585", "375", "346", "160", "601", "300", "0",
"726", "487", "241", "655", "425", "460", "450", "567", "550", "479", "564", "637", "252", "514", "453", "736", "573", "303",
"731", "401", "536", "453", "626", "490", "404", "510", "635", "516", "389", "413", "598", "690", "518", "747", "282", "604",
"831", "500", "512", "509", "644", "195", "692", "574", "392", "576", "522", "584", "528", "480", "795"}
Dim Dinajpur() As String = {"338", "467", "651", "584", "505", "541", "141", "416", "450", "578", "331", "432", "726", "0",
"378", "485", "120", "310", "463", "469", "313", "374", "519", "329", "89", "595", "434", "342", "134", "283", "472", "129",
"428", "355", "329", "321", "509", "362", "320", "127", "408", "352", "358", "207", "216", "359", "57", "494", "266", "94",
"540", "489", "347", "205", "629", "78", "446", "439", "329", "213", "603", "547", "246", "58"}
Dim Faridpur() As String = {"101", "140", "413", "209", "131", "167", "239", "206", "144", "339", "118", "193", "487",
"378", "0", "246", "308", "128", "89", "260", "235", "96", "144", "81", "290", "356", "155", "214", "388", "96", "171", "383",
"73", "52", "65", "143", "300", "102", "203", "268", "100", "109", "148", "171", "264", "239", "399", "206", "145", "484",
"166", "147", "31", "217", "390", "345", "168", "96", "250", "190", "393", "337", "148", "447"}
Dim Feni() As String = {"149", "270", "167", "303", "224", "260", "344", "134", "105", "93", "360", "58", "241", "485",
"246", "0", "414", "183", "219", "209", "325", "309", "237", "323", "396", "110", "273", "212", "495", "332", "62", "490",
"160", "294", "212", "385", "249", "163", "269", "394", "275", "147", "172", "357", "449", "277", "505", "41", "363", "590",
"259", "271", "267", "403", "144", "451", "332", "151", "334", "280", "343", "287", "239", "553"}
Dim Gaibandha() As String = {"268", "397", "581", "513", "435", "471", "71", "345", "380", "507", "261", "361", "655",
"120", "308", "414", "0", "239", "392", "399", "243", "304", "448", "259", "77", "524", "363", "271", "120", "212", "402",
"115", "358", "285", "259", "250", "439", "292", "249", "117", "338", "282", "287", "137", "229", "288", "131", "423", "195",
"216", "470", "419", "277", "182", "558", "76", "376", "368", "258", "142", "532", "476", "176", "179"}
Dim Gazipur() As String = {"37", "215", "350", "284", "206", "242", "169", "114", "149", "276", "237", "130", "425", "310",
"128", "183", "239", "0", "164", "168", "151", "216", "219", "200", "221", "294", "217", "99", "320", "184", "171", "315",
"127", "172", "64", "241", "208", "61", "93", "219", "167", "51", "56", "182", "274", "129", "330", "193", "188", "415", "241",
"222", "119", "228", "328", "276", "287", "138", "159", "105", "301", "246", "64", "378"}
Dim Gopalganj() As String = {"127", "52", "386", "196", "117", "153", "324", "231", "130", "312", "171", "218", "460",
"463", "89", "219", "392", "164", "0", "285", "305", "89", "91", "135", "374", "329", "54", "239", "473", "180", "157", "468",
"59", "105", "149", "197", "325", "127", "249", "353", "55", "134", "173", "256", "348", "285", "483", "192", "230", "568",
"152", "59", "116", "302", "363", "429", "114", "82", "314", "260", "418", "363", "219", "532"}
Dim Habiganj() As String = {"163", "336", "376", "406", "327", "363", "328", "76", "218", "302", "373", "156", "450",
"469", "260", "209", "399", "168", "285", "0", "257", "322", "340", "336", "380", "319", "338", "132", "479", "322", "239",
"474", "248", "308", "202", "380", "63", "176", "200", "378", "289", "161", "121", "341", "433", "197", "490", "218", "347",
"574", "362", "343", "257", "387", "353", "435", "398", "259", "258", "265", "137", "81", "223", "538"}
Dim Jamalpur() As String = {"179", "357", "492", "426", "348", "384", "172", "203", "291", "418", "276", "272", "567",
"313", "235", "325", "243", "151", "305", "257", "0", "319", "361", "274", "224", "436", "378", "125", "323", "227", "313",
"318", "269", "278", "170", "266", "297", "203", "57", "222", "309", "193", "198", "185", "278", "96", "334", "335", "191",
"418", "383", "364", "226", "231", "470", "279", "391", "280", "16", "109", "391", "335", "87", "382"}
Dim Jessore() As String = {"164", "93", "476", "209", "168", "204", "235", "269", "195", "402", "82", "256", "550", "374",
"96", "309", "304", "216", "89", "322", "319", "0", "148", "46", "286", "419", "59", "277", "384", "92", "222", "379", "124",
"44", "152", "108", "362", "165", "291", "264", "34", "172", "211", "167", "259", "327", "395", "257", "141", "480", "200",
"115", "119", "213", "453", "340", "72", "147", "334", "219", "456", "400", "236", "443"}
Dim Jhalokathi() As String = {"182", "100", "404", "95", "19", "55", "379", "287", "149", "330", "231", "215", "479", "519",
"144", "237", "448", "219", "91", "340", "361", "148", "0", "194", "430", "348", "90", "294", "528", "236", "176", "523", "77",
"193", "205", "256", "380", "183", "305", "408", "162", "190", "229", "311", "404", "340", "539", "210", "285", "624", "52",
"33", "171", "357", "382", "485", "145", "101", "370", "316", "474", "418", "275", "587"}
Dim Jhenaidah() As String = {"178", "139", "490", "286", "207", "243", "190", "283", "220", "416", "37", "270", "564",
"329", "81", "323", "259", "200", "135", "336", "274", "46", "194", "0", "240", "433", "105", "281", "339", "46", "248", "334",
"149", "28", "137", "62", "376", "179", "280", "219", "79", "186", "225", "122", "214", "319", "350", "282", "96", "434", "242",
"161", "103", "168", "467", "295", "118", "173", "289", "173", "470", "414", "207", "398"}
Dim Joypurhat() As String = {"249", "378", "563", "495", "416", "452", "52", "327", "361", "489", "243", "343", "637", "89",
"290", "396", "77", "221", "374", "380", "224", "286", "430", "240", "0", "506", "345", "253", "157", "194", "384", "152",
"340", "267", "241", "232", "420", "274", "231", "39", "319", "264", "269", "118", "128", "270", "103", "405", "177", "181",
"451", "401", "259", "116", "540", "113", "358", "350", "240", "124", "514", "458", "158", "144"}
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Dim Khagrachhari() As String = {"259", "381", "133", "413", "334", "370", "454", "244", "215", "117", "470", "169", "252", "595", "356", "110", "524", "294", "329", "319", "436", "419", "348", "433", "506", "0", "383", "322", "605", "442", "172", "600", "270", "405", "322", "495", "359", "273", "379", "504", "385", "258", "282", "467", "559", "387", "615", "151", "473", "700", "369", "381", "378", "513", "70", "561", "442", "261", "445", "390", "453", "397", "349", "664"}
Dim Khulna() As String = {"180", "34", "439", "150", "109", "145", "295", "285", "184", "366", "142", "272", "514", "434", "155", "273", "363", "217", "54", "338", "378", "59", "90", "105", "345", "383", "0", "292", "444", "151", "211", "439", "113", "103", "211", "167", "378", "181", "302", "323", "90", "188", "227", "227", "319", "338", "454", "246", "200", "539", "141", "56", "177", "272", "417", "400", "59", "136", "394", "278", "472", "416", "273", "502"}
Dim Kishoreganj() As String = {"117", "290", "379", "360", "281", "317", "201", "79", "221", "305", "305", "159", "453", "342", "214", "212", "271", "99", "239", "132", "125", "277", "294", "281", "253", "322", "292", "0", "352", "256", "242", "347", "203", "253", "145", "294", "172", "130", "68", "251", "243", "115", "75", "214", "306", "65", "363", "221", "220", "447", "316", "297", "200", "260", "356", "308", "352", "213", "125", "138", "266", "210", "116", "411"}
Dim Kurigram() As String = {"348", "477", "661", "594", "515", "551", "151", "425", "460", "588", "341", "441", "736", "134", "388", "495", "120", "320", "473", "479", "323", "384", "528", "339", "157", "605", "444", "352", "0", "293", "482", "29", "438", "365", "339", "331", "519", "372", "329", "197", "418", "362", "368", "217", "309", "368", "111", "504", "276", "196", "550", "499", "357", "263", "639", "56", "456", "449", "339", "223", "613", "557", "256", "159"}
Dim Kushtia() As String = {"183", "184", "499", "301", "222", "258", "144", "268", "235", "425", "49", "279", "573", "283", "96", "332", "212", "184", "180", "322", "227", "92", "236", "46", "194", "442", "151", "256", "293", "0", "262", "288", "164", "73", "120", "58", "362", "194", "234", "172", "125", "199", "210", "76", "168", "273", "303", "297", "49", "388", "257", "207", "65", "122", "476", "249", "164", "188", "243", "127", "455", "400", "161", "351"}
Dim Lakshmipur() As String = {"137", "209", "229", "241", "162", "198", "331", "164", "43", "155", "284", "89", "303", "472", "171", "62", "402", "171", "157", "239", "313", "222", "176", "248", "384", "172", "211", "242", "482", "262", "0", "477", "98", "219", "200", "310", "279", "150", "257", "381", "189", "135", "160", "338", "430", "292", "493", "35", "312", "578", "197", "209", "198", "384", "206", "438", "270", "89", "322", "268", "373", "317", "227", "541"}
Dim Lalmonirhat() As String = {"343", "472", "656", "589", "510", "546", "146", "420", "455", "583", "336", "436", "731", "129", "383", "490", "115", "315", "468", "474", "318", "379", "523", "334", "152", "600", "439", "347", "29", "288", "477", "0", "433", "360", "334", "326", "514", "367", "324", "192", "413", "357", "363", "212", "304", "363", "106", "499", "271", "191", "545", "494", "352", "258", "634", "51", "451", "444", "334", "218", "608", "552", "251", "154"}
Dim Madaripur() As String = {"90", "110", "327", "143", "64", "100", "287", "195", "71", "253", "186", "138", "401", "428", "73", "160", "358", "127", "59", "248", "269", "124", "77", "149", "340", "270", "113", "203", "438", "164", "98", "433", "0", "121", "133", "212", "289", "91", "213", "337", "90", "98", "137", "240", "332", "249", "449", "133", "214", "534", "99", "111", "100", "286", "304", "395", "172", "23", "278", "224", "382", "326", "183", "497"}
Dim Magura() As String = {"150", "136", "461", "258", "179", "215", "217", "254", "192", "387", "65", "241", "536", "355", "52", "294", "285", "172", "105", "308", "278", "44", "193", "28", "267", "405", "103", "253", "365", "73", "219", "360", "121", "0", "108", "91", "348", "151", "247", "245", "50", "158", "197", "148", "241", "283", "376", "254", "122", "461", "214", "158", "75", "194", "439", "322", "115", "144", "294", "200", "442", "386", "192", "424"}
Dim Manikganj() As String = {"63", "200", "379", "270", "191", "227", "188", "148", "177", "305", "173", "159", "453", "329", "65", "212", "259", "64", "149", "202", "170", "152", "205", "137", "241", "322", "211", "145", "339", "120", "200", "334", "133", "108", "0", "178", "242", "89", "138", "239", "156", "80", "90", "202", "294", "174", "350", "221", "169", "435", "226", "208", "55", "248", "356", "296", "224", "156", "186", "125", "335", "280", "84", "398"}
Dim Meherpur() As String = {"240", "201", "552", "348", "270", "306", "182", "326", "283", "478", "26", "332", "626", "321", "143", "385", "250", "241", "197", "380", "266", "108", "256", "62", "232", "495", "167", "294", "331", "58", "310", "326", "212", "91", "178", "0", "420", "241", "272", "211", "142", "248", "268", "114", "206", "311", "342", "345", "88", "426", "305", "223", "122", "160", "529", "287", "180", "235", "281", "165", "513", "457", "199", "390"}
Dim Moulvibazar() As String = {"203", "376", "416", "446", "367", "403", "368", "116", "258", "342", "413", "196", "490", "509", "300", "249", "439", "208", "325", "63", "297", "362", "380", "376", "420", "359", "378", "172", "519", "362", "279", "514", "289", "348", "242", "420", "0", "216", "240", "418", "329", "201", "161", "381", "474", "237", "530", "258", "387", "614", "402", "383", "297", "427", "393", "475", "438", "299", "298", "305", "117", "58", "264", "578"}
Dim Munshiganj() As String = {"27", "179", "329", "248", "169", "205", "221", "123", "128", "256", "216", "109", "404", "362", "102", "163", "292", "61", "127", "176", "203", "165", "183", "179", "274", "273", "181", "130", "372", "194", "150", "367", "91", "151", "89", "241", "216", "0", "147", "271", "131", "15", "65", "234", "327", "182", "383", "172", "240", "468", "204", "186", "129", "280", "307", "328", "240", "101", "212", "158", "310", "254", "117", "431"}
Dim Mymensingh() As String = {"122", "300", "436", "370", "291", "327", "178", "147", "234", "362", "282", "216", "510", "320", "203", "269", "249", "93", "249", "200", "57", "291", "305", "280", "231", "379", "302", "68", "329", "234", "257", "324", "213", "247", "138", "272", "240", "147", "0", "229", "253", "137", "135", "192", "284", "39", "340", "278", "198", "425", "326", "308", "194", "238", "413", "286", "362", "223", "69", "115", "334", "278", "94", "388"}
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Dim Naogaon() As String = {"247", "357", "561", "473", "395", "431", "50", "325", "359", "487", "221", "341", "635", "127", "268", "394", "117", "219", "353", "378", "222", "264", "408", "219", "39", "504", "323", "251", "197", "172", "381", "192", "337", "245", "239", "211", "418", "271", "229", "0", "298", "261", "267", "97", "98", "268", "142", "403", "156", "219", "430", "379", "237", "78", "538", "153", "336", "348", "238", "122", "512", "456", "156", "182"}
Dim Narail() As String = {"130", "107", "442", "227", "148", "184", "269", "235", "162", "368", "116", "222", "516", "408", "100", "275", "338", "167", "55", "289", "309", "34", "162", "79", "319", "385", "90", "243", "418", "125", "189", "413", "90", "50", "156", "142", "329", "131", "253", "298", "0", "138", "177", "201", "293", "289", "429", "223", "175", "513", "183", "114", "122", "247", "419", "374", "103", "114", "318", "252", "422", "366", "223", "477"}
Dim Narayanganj() As String = {"17", "186", "314", "255", "177", "213", "211", "107", "113", "240", "223", "94", "389", "352", "109", "147", "282", "51", "134", "161", "193", "172", "190", "186", "264", "258", "188", "115", "362", "199", "135", "357", "98", "158", "80", "248", "201", "15", "137", "261", "138", "0", "49", "224", "317", "172", "373", "157", "230", "458", "212", "193", "135", "270", "292", "318", "247", "108", "202", "148", "294", "239", "107", "421"}
Dim Narsingdi() As String = {"51", "225", "339", "294", "215", "251", "217", "67", "137", "265", "262", "119", "413", "358", "148", "172", "287", "56", "173", "121", "198", "211", "229", "225", "269", "282", "227", "75", "368", "210", "160", "363", "137", "197", "90", "268", "161", "65", "135", "267", "177", "49", "0", "230", "322", "140", "378", "181", "236", "463", "250", "232", "146", "276", "316", "324", "286", "147", "200", "153", "254", "198", "112", "426"}
Dim Natore() As String = {"210", "260", "524", "377", "298", "334", "68", "288", "311", "450", "124", "304", "598", "207", "171", "357", "137", "182", "256", "341", "185", "167", "311", "122", "118", "467", "227", "214", "217", "76", "338", "212", "240", "148", "202", "114", "381", "234", "192", "97", "201", "224", "230", "092", "231", "228", "366", "59", "312", "333", "282", "140", "46", "501", "173", "239", "263", "201", "85", "475", "419", "119", "276"}
Dim Nawabganj() As String = {"302", "352", "616", "469", "390", "426", "160", "380", "403", "542", "217", "396", "690", "216", "264", "449", "229", "274", "348", "433", "278", "259", "404", "214", "128", "559", "319", "306", "309", "168", "430", "304", "332", "241", "294", "206", "474", "327", "284", "98", "293", "317", "322", "92", "0", "323", "230", "458", "151", "308", "425", "375", "232", "46", "593", "265", "332", "356", "293", "177", "567", "511", "211", "271"}
Dim Netrakona() As String = {"158", "336", "444", "406", "327", "363", "217", "144", "270", "370", "321", "224", "518", "359", "239", "277", "288", "129", "285", "197", "96", "327", "340", "319", "270", "387", "338", "65", "368", "273", "292", "363", "249", "283", "174", "311", "237", "182", "39", "268", "289", "172", "140", "231", "323", "0", "379", "286", "237", "464", "362", "343", "230", "277", "421", "325", "398", "259", "96", "154", "331", "275", "133", "427"}
Dim Nilphamari() As String = {"359", "488", "672", "604", "526", "562", "162", "436", "471", "598", "352", "452", "747", "57", "399", "505", "131", "330", "483", "490", "334", "395", "539", "350", "103", "615", "454", "363", "111", "303", "493", "106", "449", "376", "350", "342", "530", "383", "340", "142", "429", "373", "378", "228", "230", "379", "0", "514", "286", "68", "561", "510", "368", "274", "649", "55", "467", "459", "349", "233", "623", "567", "267", "62"}
Dim Noakhali() As String = {"158", "244", "208", "276", "197", "233", "353", "143", "78", "134", "319", "67", "282", "494", "206", "41", "423", "193", "192", "218", "335", "257", "210", "282", "405", "151", "246", "221", "504", "297", "35", "499", "133", "254", "221", "345", "258", "172", "278", "403", "223", "157", "181", "366", "458", "286", "514", "0", "347", "599", "232", "244", "233", "412", "185", "460", "305", "124", "344", "289", "352", "296", "248", "562"}
Dim Pabna() As String = {"216", "234", "530", "350", "272", "308", "127", "294", "285", "456", "98", "310", "604", "266", "145", "363", "195", "188", "230", "347", "191", "141", "285", "96", "177", "473", "200", "220", "276", "49", "312", "271", "214", "122", "169", "88", "387", "240", "198", "156", "175", "230", "236", "59", "151", "237", "286", "347", "0", "371", "307", "256", "114", "105", "507", "232", "213", "237", "207", "91", "481", "425", "125", "335"}
Dim Panchagarh() As String = {"443", "572", "757", "689", "610", "646", "246", "521", "555", "683", "437", "537", "831", "94", "484", "590", "216", "415", "568", "574", "418", "480", "624", "434", "181", "700", "539", "447", "196", "388", "578", "191", "534", "461", "435", "426", "614", "468", "425", "219", "513", "458", "463", "312", "308", "464", "68", "599", "371", "0", "645", "595", "453", "296", "734", "140", "552", "544", "434", "318", "708", "652", "352", "37"}
Dim Patuakhali() As String = {"204", "151", "426", "44", "41", "77", "401", "308", "170", "352", "279", "237", "500", "540", "166", "259", "470", "241", "152", "362", "383", "200", "52", "242", "451", "369", "141", "316", "550", "257", "197", "545", "99", "214", "226", "305", "402", "204", "326", "430", "183", "212", "250", "333", "425", "362", "561", "232", "307", "645", "0", "85", "193", "379", "403", "506", "197", "122", "392", "337", "495", "440", "296", "609"}
Dim Pirojpur() As String = {"185", "66", "438", "96", "53", "89", "350", "290", "182", "364", "198", "249", "512", "489", "147", "271", "419", "222", "59", "343", "364", "115", "33", "161", "401", "381", "56", "297", "499", "207", "209", "494", "111", "158", "208", "223", "383", "186", "308", "379", "114", "193", "232", "282", "375", "343", "510", "244", "256", "595", "85", "0", "174", "328", "415", "455", "112", "134", "373", "334", "477", "421", "278", "558"}
Dim Rajbari() As String = {"118", "167", "434", "236", "158", "194", "208", "204", "171", "360", "108", "214", "509", "347", "31", "267", "277", "119", "116", "257", "226", "119", "171", "103", "259", "378", "177", "200", "357", "65", "198", "352", "100", "75", "55", "122", "297", "129", "194", "237", "122", "135", "146", "140", "232", "230", "368", "233", "114", "453", "193", "174", "0", "186", "412", "313", "190", "123", "241", "180", "391", "335", "139", "416"}
Dim Rajshahi() As String = {"256", "306", "569", "423", "344", "380", "114", "334", "357", "496", "170", "350", "644", "205", "217", "403", "182", "228", "302", "387", "231", "213", "357", "168", "116", "513", "272", "260", "263", "122", "384", "258", "286", "194", "248", "160", "427", "280", "238", "78", "247", "270", "276", "46", "46", "277", "274", "412", "105", "296", "379", "328", "186", "0", "547", "219", "285", "309", "247", "131", "521", "465", "164", "260"}
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    Dim Rangamati() As String = {"293", "415", "74", "447", "368", "404", "488", "278", "249", "78", "504", "203", "195",
"629", "390", "144", "558", "328", "363", "353", "470", "453", "382", "467", "540", "70", "417", "356", "639", "476", "206",
"634", "304", "439", "356", "529", "393", "307", "413", "538", "419", "292", "316", "501", "593", "421", "649", "185", "507",
"734", "403", "415", "412", "547", "0", "595", "476", "295", "479", "424", "487", "431", "383", "698"}
    Dim Rangpur() As String = {"304", "433", "618", "550", "471", "507", "107", "382", "416", "544", "297", "398", "692", "78",
"345", "451", "76", "276", "429", "435", "279", "340", "485", "295", "113", "561", "400", "308", "56", "249", "438", "51",
"395", "322", "296", "287", "475", "328", "286", "153", "374", "318", "324", "173", "265", "325", "55", "460", "232", "140",
"506", "455", "313", "219", "595", "0", "412", "405", "295", "179", "569", "513", "213", "103"}
    Dim Satkhira() As String = {"240", "94", "499", "206", "164", "200", "307", "344", "243", "425", "154", "331", "574", "446",
"168", "332", "376", "287", "114", "398", "391", "72", "145", "118", "358", "442", "59", "352", "456", "164", "270", "451",
"172", "115", "224", "180", "438", "240", "362", "336", "103", "247", "286", "239", "332", "398", "467", "305", "213", "552",
"197", "112", "190", "285", "476", "412", "0", "195", "406", "291", "531", "476", "307", "515"}
    Dim Shariatpur() As String = {"101", "134", "317", "166", "87", "123", "298", "205", "62", "244", "209", "128", "392",
"439", "96", "151", "368", "138", "82", "259", "280", "147", "101", "173", "350", "261", "136", "213", "449", "188", "89",
"444", "23", "144", "156", "235", "299", "101", "223", "348", "114", "108", "147", "263", "356", "259", "459", "124", "237",
"544", "122", "134", "123", "309", "295", "405", "195", "0", "289", "234", "392", "337", "193", "507"}
    Dim Sherpur() As String = {"188", "366", "501", "435", "357", "393", "188", "204", "300", "427", "291", "281", "576",
"329", "250", "334", "258", "159", "314", "258", "16", "334", "370", "289", "240", "445", "394", "125", "339", "243", "322",
"334", "278", "294", "186", "281", "298", "212", "69", "238", "318", "202", "200", "201", "293", "96", "349", "344", "207",
"434", "392", "373", "241", "247", "479", "295", "406", "289", "0", "124", "391", "335", "103", "397"}
    Dim Sirajganj() As String = {"134", "311", "447", "381", "302", "338", "72", "211", "246", "373", "176", "227", "522",
"213", "190", "280", "142", "105", "260", "265", "109", "219", "316", "173", "124", "390", "278", "138", "223", "127", "268",
"218", "224", "200", "125", "165", "305", "158", "115", "122", "252", "148", "153", "85", "177", "154", "233", "289", "91",
"318", "337", "334", "180", "131", "424", "179", "291", "234", "124", "0", "398", "342", "42", "281"}
    Dim Sunamganj() As String = {"296", "470", "509", "539", "460", "496", "462", "209", "351", "436", "507", "290", "584",
"603", "393", "343", "532", "301", "418", "137", "391", "456", "474", "470", "514", "453", "472", "266", "613", "455", "373",
"608", "382", "442", "335", "513", "117", "310", "334", "512", "422", "294", "254", "475", "567", "331", "623", "352", "481",
"708", "495", "477", "391", "521", "487", "569", "531", "392", "391", "398", "0", "68", "357", "671"}
    Dim Sylhet() As String = {"241", "414", "454", "483", "405", "441", "406", "153", "295", "380", "451", "234", "528", "547",
"337", "287", "476", "246", "363", "81", "335", "400", "418", "414", "458", "397", "416", "210", "557", "400", "317", "552",
"326", "386", "280", "457", "58", "254", "278", "456", "366", "239", "198", "419", "511", "275", "567", "296", "425", "652",
"440", "421", "335", "465", "431", "513", "476", "337", "335", "342", "68", "0", "301", "616"}
    Dim Tangail() As String = {"92", "270", "406", "340", "261", "297", "105", "170", "204", "332", "209", "186", "480", "246",
"148", "239", "176", "64", "219", "223", "87", "236", "275", "207", "158", "349", "273", "116", "256", "161", "227", "251",
"183", "192", "84", "199", "264", "117", "94", "156", "223", "107", "112", "119", "211", "133", "267", "248", "125", "352",
"296", "278", "139", "164", "383", "213", "307", "193", "103", "42", "357", "301", "0", "315"}
    Dim Thakurgaon() As String = {"407", "536", "720", "652", "574", "610", "210", "484", "519", "646", "400", "500", "795",
"58", "447", "553", "179", "378", "532", "538", "382", "443", "587", "398", "144", "664", "502", "411", "159", "351", "541",
"154", "497", "424", "398", "390", "578", "431", "388", "182", "477", "421", "426", "276", "271", "427", "62", "562", "335",
"37", "609", "558", "416", "260", "698", "103", "515", "507", "397", "281", "671", "616", "315", "0"}
```

এবার ডিক্লায়র করবেন ইনডেক্স ও শহরগুলোর নাম:

```
 Dim ProthomIndex As Integer : Dim DithioIndex As Integer 'declare globals for index value of listBox
    'load names of cities in an array in the memory.
    Dim NamesOfCity() As String = {"Dhaka", "Bagerhat", "Bandarban", "Barguna", "Barisal", "Bhola", "Bogra",
"Brahmanbaria", "Chandpur", "Chittagong", "Chuadanga", "Comilla", "Cox'sBaxar", _
                         "Dinajpur", "Faridpur", "Feni", "Gaibandha", "Gazipur", "Gopalganj", "Habiganj", "Jamalpur", "Jessore",
"Jhalokathi", "Jhenaidah", "Joypurhat", _
                         "Khagrachhari", "Khulna", "Kishoreganj", "Kurigram", "Kushtia", "Lakshmipur", "Lalmonirhat",
"Madaripur", "Magura", "Manikganj", "Meherpur", _
                         "Moulvibazar", "Munshiganj", "Mymensingh", "Naogaon", "Narail", "Narayanganj", "Narsingdi",
"Natore", "Nawabganj", "Netrakona", "Nilphamari", _
                         "Noakhali", "Pabna", "Panchagarh", "Patuakhali", "Pirojpur", "Rajbari", "Rajshahi", "Rangamati",
"Rangpur", "Satkhira", "Shariatpur", "Sherpur", _
                         "Sirajganj", "Sunamganj", "Sylhet", "Tangail", "Thakurgaon"}
```

ব্যস! কষ্ট ও সতর্কতাসহ উপরোক্ত তথ্যাদি আপনাকে টাইপ করতেই হবে। প্রোগ্রামারকে কখনো লেজি হতে নেই!

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

ডিক্লারেশন শেষে ফর্মের ক্লাসের ভেতর প্রথম সাবরুটিন হবে ফর্মের লোড ইভেন্ট। ফর্মে ডবল ক্লিক করলেই এই লোড ইভেন্ট সাবটি দেখাবে। এতে যা লিখবেন তাহলো:-

```
' This load sub populates two ListBoxes with the names of Cities.
 Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
       Label1.Text = "" : Label5.Text = "" : Label6.Text = "" 'we don't need those names
       ' ==== dimension array for display in the list boxes. Remember, this sub is executed only once. ===
       Dim a As Integer ' a is loop control integer
       For a = 0 To 63
          ListBox1.Items.Add(NamesOfCity(a)) '0 through 63 = 64 dirstict names in the declared array.
       Next a
       For a = 0 To 63
          ListBox2.Items.Add(NamesOfCity(a)) '0 through 63 = 64 dirstict names in the declared array.
       Next a
 End Sub
```

উপরের কোড ব্লকের কাজ হলো উভয় লিস্টবাক্সে ৬৪টি জিলা শহরের নাম লিখা। এবার আমরা প্রস্তুত হয়েছি ব্যবহারকারীর চাহিদানুযায়ী দূরত্ব প্রদর্শন করতে। এ লক্ষ্যর্জনে প্রয়োজন হবে নিচের দু'টি সাবরুটিন ও দু'টি ফাংশন।

প্রথম সাবরুটিন: লিস্ট বাক্সে ডবল ক্লিক করলেই এই সাবটি দেখাবে। এতে যা লিখবেন তাহলো:-

```
'whatever selected is displayed in the labels
Private Sub ListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ListBox1.SelectedIndexChanged
      Label1.Text = ListBox1.SelectedItem 'get the selected text and display in label
      Label6.Text = "is: "
      Label5.Text = ""
      Label8.Text = "Equal: "
End Sub
```

আমরা জেনে নিলাম কোন্ শহর থেকে দূরত্ব জানার ইচ্ছা করছেন ব্যবহারকারী। এবার জানতে হবে কোন্ শহর পর্যন্ত দূরত্ব জানবেন তিনি। এছাড়া দ্বিতীয় এ তথ্যটি জানার সাথে সাথে আমরা চাই দূরত্বও প্রদর্শিত হবে- যা আমাদের প্রোগ্রামের মূল লক্ষ্য। দূরত্ব কিলোমিটার ও মাইলের এককে হতে হবে। এই কাজগুলোর জন্য দরকার নিচের দ্বিতীয় অপেক্ষাকৃত দীর্ঘ সাবরুটিনটি। এটাও লিস্ট বাক্সের (ListBox2) সিলেক্টেট ইনডেক্স পরিবর্তন ইভেন্ট সাবরুটিন। লিস্ট বাক্সে ডবল ক্লিক করলেই এটি প্রদর্শিত হবে। আপনি যা লিখবেন তাহলো:-

```
 'main display in this event
Private Sub ListBox2_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
ListBox2.SelectedIndexChanged
     Dim IndexA As Integer 'to get from index
     Dim IndexB As Integer 'to get to index
     Dim TextSel1 As String 'from city name
     Dim TextSel2 As String 'to city name

     If ListBox1.SelectedItem = "" Or ListBox2.SelectedItem = "" Then Exit Sub 'make sure selection is true
     IndexA = ListBox1.SelectedIndex 'get the index as number
     IndexB = ListBox2.SelectedIndex 'get the index as number
     TextSel1 = ListBox1.SelectedItem 'from city name
     TextSel2 = ListBox2.SelectedItem 'to city name
             'MsgBox(IndexA & " : " & IndexB)
     Label5.Text = TextSel2 'to city name into display
```

পরের পৃষ্ঠায় ... ...

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    '=== special case Dhaka. ref point
   If IndexA = 0 Then
       If IndexB = 0 Then 'Dhaka to Dhaka chosen
          Label6.Text = "is: " & 0 & " km"
          Label8.Text = "Equal: " & 0 & " Miles"
          Exit Sub
       End If
       Label6.Text = "is: " & FindDisanceZero(IndexB) & " km" 'If both condition results FALSE, this func. called
   End If
    '============= check from which city chosen ========
   Label6.Text = "is: " & ManyatOnce() & " km" 'call function and display result
    '==================== get distance in miles and display'
   Dim KMvalue As Integer = Mid(Label6.Text, 5, (Len(Label6.Text) - 7)) 'get km value to a variable as number
   Label8.Text = "Equal: " & Math.Truncate(KMvalue * 0.621371) & " M" 'convert to miles and display
   ListBox1.SelectedItem = ""
   ListBox2.SelectedItem = ""
 End Sub
```

লাইন বাই লাইন ব্যাখ্যা: প্রায় প্রতিটি লাইনের পাশে (সবুজ রঙে) কমেন্ট করেছি বুঝার জন্য। এরপরও মনে করছি বুঝিয়ে দেওয়া আপনার জন্য সুবিধে হবে।

ক. প্রথম চার লাইনে চারটি ভেরিয়েবল ডিক্লায়ার করেছি। এর দু'টি ইন্টেজার ও দু'টি স্ট্রিং টাইপ। এগুলোতে যথাক্রমে লিস্ট বাক্সের ইনডেক্স সংখ্যা ও সিলেক্টেড টেক্সট সংরক্ষিত থাকবে।
খ. পঞ্চম লাইন দ্বারা নিশ্চিত করা হয়েছে, ব্যবহারকারী উভয় লিস্ট বাক্সে কোনো শহরের নাম সিলেক্ট করেছেন। না করলে প্রোগ্রাম বন্ধ হয়ে যাবে। এমতাবস্থায় সাবরুটিন এক্সেকিউশন বন্ধ করা জরুরী- যা এই লাইনটি নিশ্চিত করে।
ঘ. এর পরের চারটি লাইন দ্বারা প্রথম চার লাইনে ডিক্লায়ার করা ভেরিয়েবলগুলোতে ভ্যালু সংরক্ষণ করেছি। অর্থাৎ উভয় লিস্ট বাক্সের ইনডেক্ট সংখ্যা ও শহরের নাম ভেরিয়েবলে নিয়ে এসেছি। আগের পৃষ্ঠার শেষ লাইনটি দ্বারা আমরা দ্বিতীয় লিস্ট বাক্সের সিলেক্টেট টেক্সট তথা শহরের নাম লেবেলে এনে প্রদর্শন করেছি। এবার আমাদের কাজ হলো সঠিক দূরত্ব কিলোমিটার ও মাইল এককে প্রদর্শন করা।
ঙ. দূরত্ব প্রদর্শনের জন্য আমরা একই ইভেন্টই ব্যবহার করবো। অর্থাৎ ব্যবহারকারী দেখবেন দ্বিতীয় লিস্ট বাক্সে ক্লিক করলেই দূরত্ব প্রদর্শিত হচ্ছে। এ কাজের জন্য প্রথমত বিশেষ কেইস বিবেচনা করতে হবে- তাহলো ঢাকা থেকে ঢাকা। উভয়টি সিলেক্ট হলে রেজাল্ট হবে ০। আমরা এ কাজের জন্য দু'টি ইফ-এন্ড ইফ [If - End If] ব্লক ব্যবহার করে চেক করেছি। দ্বিতীয় ইফ-এন্ড ইফ প্রথমটির ভেতরে নেস্ট করা হয়েছে। অন্যকথায়, যদি উভয় ইনডেক্স = ০ হয় তাহলে বুঝতে হবে ঢাকা থেকে ঢাকা চয়েজ করা হয়েছে। আর এর ফলাফল ০। সুতরাং ডিসপ্লেতে ০ প্রদর্শন করে প্রোগ্রাম এখানেই থেমে যেতে হবে। এজন্যই লেবেল ৬ ও ৮-এ ০ প্রদর্শন করে Exit Sub কোড দ্বারা সাবরুটিন থেকে বেরিয়ে আসার নির্দেশ দেওয়া হয়েছে।
চ. পরের লাইনেই মূলত দূরত্ব প্রদর্শনের নির্দেশ দেওয়া হয়েছে। এতে বলা আছে: লেবেল৬ এর টেক্স হবে:
"is: " & ManyatOnce() & " km" - যার অর্থ, উভয়দিকে দু'টি শব্দ [is: , km] থাকবে এবং মাঝে লিখা হবে দূরত্ব যা আসবে [ManyatOnce()] ফাংশনের ফলাফল থেকে।
ছ. পরের দু'টি লাইনের কাজ হলো কিলোমিটার একককে পরিবর্তন করে মাইল এককে রূপান্তর ও লিস্ট বাক্স ৮-এ তা প্রদর্শন করা। সুতরাং ব্যবহারকারী কিলোমিটার ও মাইল এ উভয় এককে দূরত্ব জানতে পারবেন। সর্বশেষ দু'টি লাইন দ্বারা উভয় লিস্ট বাক্সের আইটেম সিলেকশন বাতিল করা হয়েছে। ফলে পরবর্তীতে ভুল ফলাফল আসার সম্ভাবনা আর থাকবে না।
জ. এবার শেষের ফাংশন সম্পর্কে কিছু বলার আছে। প্রথমে এটি কী তা [পরের পৃষ্ঠায়] দেখে নিই।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Function ManyatOnce() 'called from above procedure. Returns correct distance between cities.
    ProthomIndex = ListBox1.SelectedIndex : DithioIndex = ListBox2.SelectedIndex 'get index of selected index of listBox
    If ProthomIndex = 0 Then Return Dhaka(DithioIndex)
    If ProthomIndex = 1 Then Return Bagerhat(DithioIndex)
    If ProthomIndex = 2 Then Return Bandarban(DithioIndex)
    If ProthomIndex = 3 Then Return Barguna(DithioIndex)
    If ProthomIndex = 4 Then Return Barisal(DithioIndex)
    If ProthomIndex = 5 Then Return Bhola(DithioIndex)
    If ProthomIndex = 6 Then Return Bogra(DithioIndex)
    If ProthomIndex = 7 Then Return Brahmanbaria(DithioIndex)
    If ProthomIndex = 8 Then Return Chandpur(DithioIndex)
    If ProthomIndex = 9 Then Return Chittagong(DithioIndex)
    If ProthomIndex = 10 Then Return Chuadanga(DithioIndex)
    If ProthomIndex = 11 Then Return Comilla(DithioIndex)
    If ProthomIndex = 12 Then Return coxsBazar(DithioIndex)
    If ProthomIndex = 13 Then Return Dinajpur(DithioIndex)
    If ProthomIndex = 14 Then Return Faridpur(DithioIndex)
    If ProthomIndex = 15 Then Return Feni(DithioIndex)
    If ProthomIndex = 16 Then Return Gaibandha(DithioIndex)
    If ProthomIndex = 17 Then Return Gazipur(DithioIndex)
    If ProthomIndex = 18 Then Return Gopalganj(DithioIndex)
    If ProthomIndex = 19 Then Return Habiganj(DithioIndex)
    If ProthomIndex = 20 Then Return Jamalpur(DithioIndex)
    If ProthomIndex = 21 Then Return Jessore(DithioIndex)
    If ProthomIndex = 22 Then Return Jhalokathi(DithioIndex)
    If ProthomIndex = 23 Then Return Jhenaidah(DithioIndex)
    If ProthomIndex = 24 Then Return Joypurhat(DithioIndex)
    If ProthomIndex = 25 Then Return Khagrachhari(DithioIndex)
    If ProthomIndex = 26 Then Return Khulna(DithioIndex)
    If ProthomIndex = 27 Then Return Kishoreganj(DithioIndex)
    If ProthomIndex = 28 Then Return Kurigram(DithioIndex)
    If ProthomIndex = 29 Then Return Kushtia(DithioIndex)
    If ProthomIndex = 30 Then Return Lakshmipur(DithioIndex)
    If ProthomIndex = 31 Then Return Lalmonirhat(DithioIndex)
    If ProthomIndex = 32 Then Return Madaripur(DithioIndex)
    If ProthomIndex = 33 Then Return Magura(DithioIndex)
    If ProthomIndex = 34 Then Return Manikganj(DithioIndex)
    If ProthomIndex = 35 Then Return Meherpur(DithioIndex)
    If ProthomIndex = 36 Then Return Moulvibazar(DithioIndex)
    If ProthomIndex = 37 Then Return Munshiganj(DithioIndex)
    If ProthomIndex = 38 Then Return Mymensingh(DithioIndex)
    If ProthomIndex = 39 Then Return Naogaon(DithioIndex)
    If ProthomIndex = 40 Then Return Narail(DithioIndex)
    If ProthomIndex = 41 Then Return Narayanganj(DithioIndex)
    If ProthomIndex = 42 Then Return Narsingdi(DithioIndex)
    If ProthomIndex = 43 Then Return Natore(DithioIndex)
    If ProthomIndex = 44 Then Return Nawabganj(DithioIndex)
    If ProthomIndex = 45 Then Return Netrakona(DithioIndex)
    If ProthomIndex = 46 Then Return Nilphamari(DithioIndex)
    If ProthomIndex = 47 Then Return Noakhali(DithioIndex)
    If ProthomIndex = 48 Then Return Pabna(DithioIndex)
    If ProthomIndex = 49 Then Return Panchagarh(DithioIndex)
    If ProthomIndex = 50 Then Return Patuakhali(DithioIndex)
    If ProthomIndex = 51 Then Return Pirojpur(DithioIndex)
    If ProthomIndex = 52 Then Return Rajbari(DithioIndex)
    If ProthomIndex = 53 Then Return Rajshahi(DithioIndex)
    If ProthomIndex = 54 Then Return Rangamati(DithioIndex)
    If ProthomIndex = 55 Then Return Rangpur(DithioIndex)
    If ProthomIndex = 56 Then Return Satkhira(DithioIndex)
    If ProthomIndex = 57 Then Return Shariatpur(DithioIndex)
    If ProthomIndex = 58 Then Return Sherpur(DithioIndex)
    If ProthomIndex = 59 Then Return Sirajganj(DithioIndex)
    If ProthomIndex = 60 Then Return Sunamganj(DithioIndex)
    If ProthomIndex = 61 Then Return Sylhet(DithioIndex)
    If ProthomIndex = 62 Then Return Tangail(DithioIndex)
    If ProthomIndex = 63 Then Return Thakurgaon(DithioIndex)
    Return 0
  End Function
End Class
```

ফাংশনটি বেশ দীর্ঘ - তাই না? আসলে এটা টাইপ করা তেমন কঠিন কিছু না। আপনি একটি টাইপ করে কপি-পেস্ট করবেন। এরপর শুধুমাত্র দু'টি শব্দ বদলাতে হবে। প্রথমত, ১ থেকে ৬৩ সংখ্যা হবে এবং দ্বিতীয়ত, ঢাকা থেকে ঠাকুরগাঁও লেখা থাকবে। তবে আপনাকে মনে রাখতে হবে:

(ক) ৬৬-টি লাইনের মধ্যে যে কোনো একটি মাত্র ভ্যালু রিটার্ন করবে। অর্থাৎ ব্যবহারকারী প্রথম লিস্ট বাক্সে যে শহরের ইনডেক্সে ক্লিক করবেন তা-ই নির্ধারণ করবে কোন্ ভ্যালু ফাংশন ফেরত দেবে।

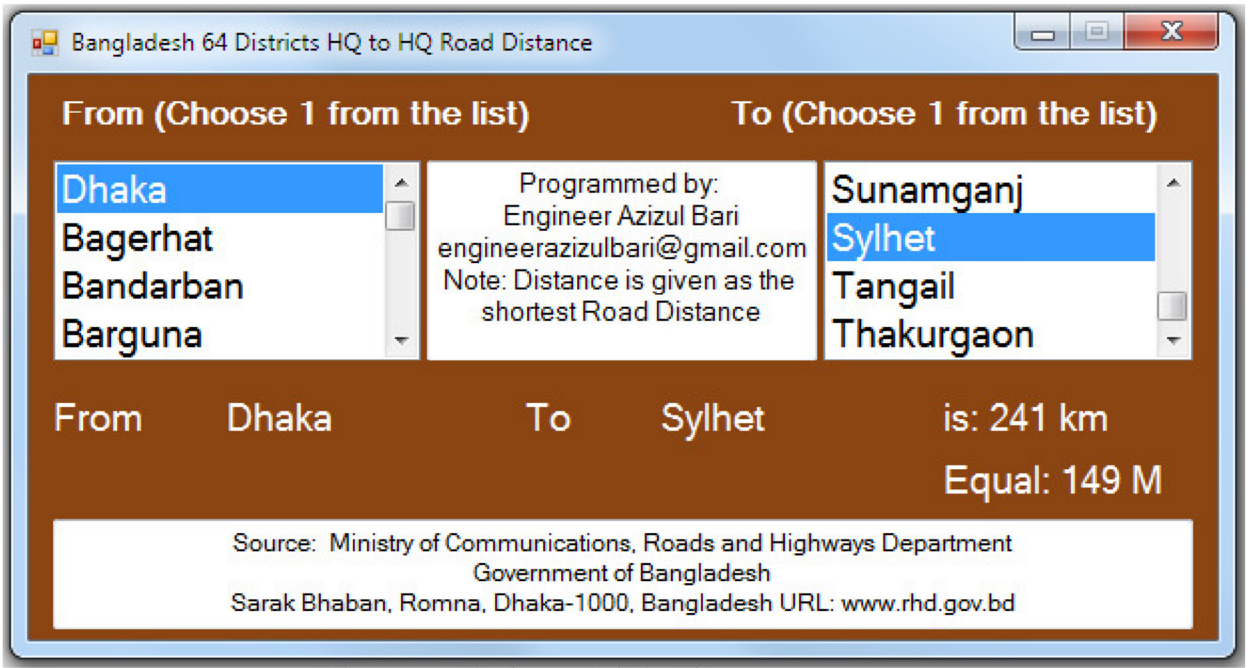
(খ) ফেরৎ ভ্যালু আসবে ফর্মের প্রথমে ডিক্লায়ার করা শহরের নামের এ্যারে ভেরিয়েবল থেকে। সুতরাং এই নামগুলোর বানান সঠিক থাকা দরকার। আমি টাইপ করার সময় ডিক্লারেশন লেখাটুকু কপি করে কাছে নিয়ে এসেছিলাম। এরপর একটা একটা করে সবগুলো কাট-পেস্ট করে বসিয়েছি। ব্যস্! ভুল হওয়ার সম্ভাবনা আর থাকে নি।

যাক, ব্যবহারযোগ্য একটি প্রোগ্রাম আমরা সৃষ্টি করতে সমর্থ হয়েছি। এখন বাকী থাকলো টেস্ট করার বিষয়টি। সুতরাং প্রোগ্রাম রান করুন এবং একটা একটা করে শহরের নামে ক্লিক করে দেখুন দূরত্ব সঠিকমতো আসছে কি না। সব ঠিকমতো কোড করা হয়ে থাকলে অবশ্যই সঠিক ফলাফল আসবে। আর কোনো সমস্যা হলে খুঁজে দেখতে হবে কোথায় ভুল-ত্রুটি হয়েছে। আর এভাবে ভুল-সংশোধনকেই বলে ডিবাগিং [Debugging]।

পরের পৃষ্ঠায় প্রোগ্রাম চলাকালের একটি চিত্র দিয়েছি। একটু দেখে নিন।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

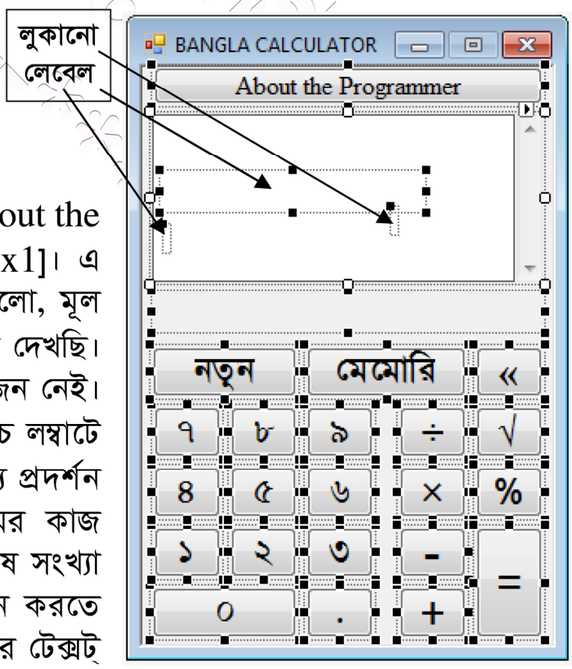


Application 1 -চলাকালীন ঢাকা থেকে সিলেটের দূরত্ব কিলোমিটার ও মাইল এককে প্রদর্শিত হচ্ছে।

Application 2 : Bangla Calculator - বাংলা ক্যালকুলেটর

উদ্দেশ্য: প্রশ্ন থাকতে পারে, বাংলা ক্যালকুলেটর- কেনো? জবাব- কেনো না? তবে আসল জবাব হলো- আমরা তো শিখতে চাই! ক্যালকুলেটর কোডিং কিভাবে করতে হয় তা জানতে চাই। আর ডিসপ্লে বাংলা সংখ্যায় হলে তো ভলোই হলো! সুতরাং আসুন, গড়ে তুলি পিসি'র জন্য একটি উন্নত বাংলা ক্যালকুলেটর এ্যাপলিকেশন।

ডিজাইন: ডানের ডিজাইনটি দেখুন। আমাদের ক্যালকুলেটরে সাধারণ পাটিগণিতের চারটি ফাংশন [+, -, ÷, x] ছাড়াও থাকবে পার্সেন্ট (%) এবং বর্গমূল (√)।



উপরে আছে প্রোগ্রামার সম্পর্কে তথ্য প্রদর্শনের একটি বোতাম [About the Programmer]। এর পরে আছে মূল প্রদর্শন জানালা [TextBox1]। এ বাক্সেই সংখ্যা প্রবেশকরণ ও প্রদর্শন ক্রিয়া সম্পাদিত হবে। এটাই হলো, মূল ডিসপ্লে। আমরা তিনটি লুকানো [label2, label3, label4] লেবেল দেখছি। এগুলোতে কিছু তথ্য সংরক্ষিত থাকবে যা ব্যবহারকারী দেখার প্রয়োজন নেই। সুতরাং এদের ভিজিবল প্রোপার্টি হবে: False। ডিসপ্লের একটু নিচে লম্বাটে একটি লেবেল [label1] আছে। এটা দ্বারা ব্যবহারকারীকে কিছু তথ্য প্রদর্শন করা হবে। বাকী সবই বোতাম। উপরের সারির তিনটি বোতামের কাজ যথাক্রমে: নতুন হিসাব, সংরক্ষিত সংখ্যা দেখা ও ডিসপ্লেতে সর্বশেষ সংখ্যা মেটানো। বাকি সব বোতামের কাজ কি শিক্ষার্থীরা সহজেই অনুমান করতে পারছেন। সুতরাং আর বেশি বলছি না। তবে মনে রাখবেন, বোতামের টেক্সট [Text] প্রোপার্টিতে যা দেখা যাচ্ছে তা ঢুকাতে হবে। অর্থাৎ ১, ২, ০, ., =, %, √ ইত্যাদি হবে বোতামগুলোর টেক্সট প্রোপার্টি।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

ওহ! আরেক কথা- ÷ লিখতে বাংলায় বিজয় কিবোর্ডে **স্ট** লিখে টাইমস্ ফন্টে পরিবর্তন করুন। √ লিখতে ম্যানু থেকে সিম্বল ইনসার্ট করতে পারেন। আর « এটার জন্য কিবোর্ডের মাধ্যমে ্র -র-ফলা প্রবেশ করে টাইমস্ ফন্টে পরিবর্তন করলেই হবে।

যাক, এই হচ্ছে ডিজাইন। চাইলে আপনি অন্যভাবেও করতে পারেন- কিছু রঙ যুক্ত করে আকর্ষণীয় করুন কিংবা ক্যালকুলেটরের সাইজ বৃদ্ধি করুন ইত্যাদি। তবে মনে রাখবেন যে অবজেক্টগুলো ফর্মে আছে তা সবই যেনো থাকে। কারণ এগুলোর নাম নিয়ে আমরা কোড করেছি।

কোডিং: ক্যালকুলেটরের কোডিং আসলে তেমন কঠিন নয়! সত্যিই বলছি। তবে হ্যাঁ, সহজ করতে যেয়ে কিছু চিন্তা-ভাবনা তো অবশ্যই করতে হবে এবং কোডিং অভিজ্ঞতা কাজে লাগানো দরকার। আসুন তাহলে কোডগুলো দেখে নিই।

আমাদের যাবতীয় কোডই শুধুমাত্র ফর্মের ক্লাস ইভেন্ট সাবরুটিনে লিখিত হয়েছে। এবার একে একে ক্লাসের ভেতরের সাবরুটিন ও ফাংশনগুলোর ওপর বিস্তারিত আলোচনার প্রয়াস পাচ্ছি।

```
Public Class Form1
    'this function tests if the display empty or not.
    'if empty returns true, otherwise false
    Function DisplayNotEmpty()
        If TextBox1.Text = "" Then
            Return True
        End If
        Return False
    End Function
```

উপরের ফাংশনের কাজ হলো মূল ডিসপ্লে বা প্রদর্শন টেক্স বাক্সে [TextBox1] সংখ্যা লিখা আছে কি না তা পরীক্ষা করে ফলাফল [True or False] রিটার্ন করা। এ ফাংশন আমরা প্রয়োজনে অন্যান্য সাবরুটিন থেকে কল করবো।

```
 'this sub is for '1-9, . and 0' entry to TextBox1 (the Display) directly
  Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress
      e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -
1)))
      ' And (sender.Text.IndexOf(".") = -1)) -this ensures . isn't entered more then once.
    End Sub
```

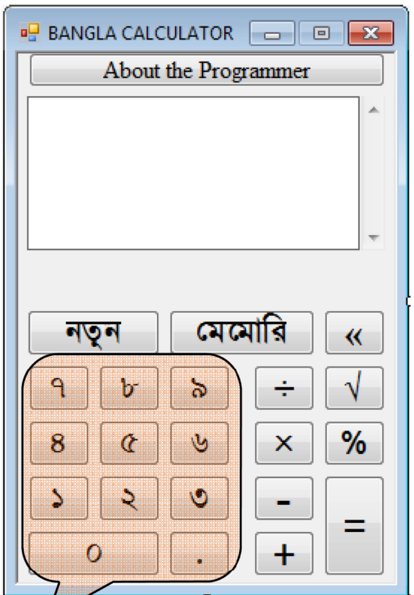
উপরের এ দ্বিতীয় সাবটি সম্পর্কে ইতোমধ্যে বলেছি। এটার কাজ হলো টেক্স বাক্সে শুধুমাত্র আস্ত কিংবা দশমিক সংখ্যা ঢুকানোর নিশ্চয়তা দেওয়া। অন্য কিছুই টাইপ করে টেক্স বাক্সে ঢুকানো যাবে না। এ কোডটি টেক্স বাক্সের KeyPress ইভেন্ট হ্যান্ডলারে দিতে হবে। এই হ্যান্ডলার পেতে প্রথমে কোড জানালার বায়ের লিস্টের সর্বশেষ আইটেম [TextBox1] চয়েজ করুন। এবার ডানের লিস্ট খুলে চয়েজ করুন কী প্রেস [KeyPress] আইটেমটি। ব্যস্! উপরোক্ত হ্যান্ডলার সাবটি এমনিতেই এসে যাবে। আপনাকে এর ভেতরে [উপরে দেখানো] একটি মাত্র লাইন লিখতে হবে।

এর পরে মোট ১১টি সাবরুটিন আছে যাদের কাজ একই ধরনের। সংখ্যা ১ থেকে ৯, ০ এবং . [দশমিক বিন্দু] টেক্স বাক্সে ঢুকানোর জন্য মোট ১১টি বোতাম আছে। ওগুলোতে ক্লিক করলেই সংশ্লিষ্ট সংখ্যা [বা একটি দশমিক বিন্দু] বাক্সে প্রদর্শিত হবে। পরের পৃষ্ঠায় এই ১১টি কোড ব্লক একত্রে দিয়েছি। দেখে দেখে টাইপ করে নিন। মনে রাখবেন, ক্লিক ইভেন্ট হ্যান্ডলার আসবে বোতামগুলো ডবল ক্লিক করার মাধ্যমে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
'this sub is for '.' entry
  Private Sub ButPoint_Click(sender As Object, e As EventArgs) Handles ButPoint.Click
     If InStr(TextBox1.Text, ".") > 0 Then Exit Sub 'make sure '.' not entered more then once.
     TextBox1.Text = TextBox1.Text & "." 'add the point after existing text
  End Sub
  'But0 - But9 codes
  Private Sub But0_Click(sender As Object, e As EventArgs) Handles But0.Click
     TextBox1.Text = TextBox1.Text & "0" 'must keep the previous number
  End Sub
  Private Sub But1_Click(sender As Object, e As EventArgs) Handles But1.Click
     TextBox1.Text = TextBox1.Text & "1"
  End Sub
  Private Sub But2_Click(sender As Object, e As EventArgs) Handles But2.Click
     TextBox1.Text = TextBox1.Text & "2"
  End Sub
  Private Sub But3_Click(sender As Object, e As EventArgs) Handles But3.Click
     TextBox1.Text = TextBox1.Text & "3"
  End Sub
  Private Sub But4_Click(sender As Object, e As EventArgs) Handles But4.Click
     TextBox1.Text = TextBox1.Text & "4"
  End Sub
  Private Sub But5_Click(sender As Object, e As EventArgs) Handles But5.Click
     TextBox1.Text = TextBox1.Text & "5"
  End Sub
  Private Sub But6_Click(sender As Object, e As EventArgs) Handles But6.Click
     TextBox1.Text = TextBox1.Text & "6"
  End Sub
  Private Sub But7_Click(sender As Object, e As EventArgs) Handles But7.Click
     TextBox1.Text = TextBox1.Text & "7"
  End Sub
  Private Sub But8_Click(sender As Object, e As EventArgs) Handles But8.Click
     TextBox1.Text = TextBox1.Text & "8"
  End Sub
  Private Sub But9_Click(sender As Object, e As EventArgs) Handles But9.Click
     TextBox1.Text = TextBox1.Text & "9"
  End Sub
```



এই ১১ বোতামের কোড উপরে দেখানো হয়েছে।

লক্ষ করুন: ButPoint বোতামের কোডের প্রথম লাইন দ্বারা নিশ্চিত করা হয়েছে, দশমিক বিন্দু [.] যাতে একবারই ঢুকানো সম্ভব হয়।

প্রত্যেক সাবরুটিনে বলা হয়েছে:
TextBox1.Text = TextBox1.Text & ...

এভাবে বলার কারণ কি জানেন? আমরা চাইনা আগের সংখ্যা মুছে যাবে। এটা নিশ্চিত হয়েছে TextBox1.Text & বলার মাধ্যমে। এরপর এসেছে সংশ্লিষ্ট সংখ্যাটি ঢুকানোর নির্দেশ। সুতরাং প্রত্যেকবার প্রথমে টেক্স বাক্সে যা আছে তা সংরক্ষণ হবে এবং এর সাথে ঢুকানো সংখ্যাটি যুক্ত হয়ে প্রদর্শিত হবে।

মনে রাখবের বোতামের মধ্যে ডবল ক্লিক করলেই ক্লিক ইভেন্ট সাবটি আসবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

পরবর্তী কোড হলো, ButNew নামক বোতামের ক্লিক ইভেন্ট। আমরা বাংলায় এর নামকরণ করেছি ‘নতুন’। এটার কাজ হলো সব মুছে নতুন হিসাব করার জন্য ক্যালকুলেটরকে প্রস্তুত করা। তবে মেমোরি মুছে যাবে না।

```
'clears all for new calculation but not the memory value kept in label2
 Private Sub ButNew_Click(sender As Object, e As EventArgs) Handles ButNew.Click
     Label3.Text = "" : TextBox1.Text = ""
     Label1.Text = "" : Label4.Text = ""
 End Sub
```

লক্ষ করুন: Label3 -তে থাকবে +, – ইত্যাদি সংকেত চিহ্ন। এটা দ্বারা বুঝা যাবে ইতোমধ্যে ব্যবহারকারী প্রথম সংখ্যা ডুকিয়ে সংকেত বোতামে ক্লিক করে নিয়েছেন কি না। কারণ, ক্লিক করে নিয়ে থাকলে তো আর পুনরায় ঢুকানোর প্রয়োজন নেই। TextBox1 -এ থাকবে প্রদর্শিত সংখ্যা। Label1 -এ থাকবে হিসাবের তথ্যাদি [এটাই হলো বোতামগুলোর উপরস্থ লম্বাটে লেবেলটি]। সবশেষে Label4-এ থকাবে ক্যালকুলেশনের জন্য ব্যবহৃত সংকেত।

```
 'gets the value from memory to Display
   Private Sub ButMem_Click(sender As Object, e As EventArgs) Handles ButMem.Click
     If Label2.Text <> "" Then TextBox1.Text = Label2.Text 'make sure it contains value before import
   End Sub
```

উপরের সাবের কাজ হলো [Label2 -তে] সংরক্ষিত মেমোরির সংখ্যা ডিসপ্লে ফিল্ডে নিয়ে আসা। লক্ষ করুন, প্রথমে টেস্ট এসেছে মেমোরিতে কিছু আছে কি না- থাকলেই তো আনা যাবে। এর পরের ৫টি সাবের কাজ প্রায় একই। এগুলো দ্বারা +, – ইত্যাদি সংকেত সংশ্লিষ্ট লেবেলে সংরক্ষণ করা হয়। লক্ষ করুন প্রতিটি সাবই `DisplayNotEmpty` নামক একটি সাবরুটিন ও `FirstValue` নামক একটি ফাংশন কল করে। প্রথম সাবটির কথা আগেই বলেছি। এর কাজ হচ্ছে ডিসপ্লেতে সংখ্যা আছে কি না তা চেক করা। আর ফাংশনটির কাজ হলো ডিসপ্লে থেকে সংখ্যা কপি করে লুকানো লেবেল [Label3 -তো] নিয়ে যাওয়া এবং ডিসপ্লে মুছে দেওয়া। আমরা এ সম্পর্কে আরো কিছু ব্যাখ্যা একটু পরই তুলে ধরবো। প্রথমে ৫টি সাব লিখে দিচ্ছি।

```
'These 5 subs entries operand signs in label4 and label1. Calls firstValue sub
   'to entry value into hidden Label3 and clears display
   'Plus Button
Private Sub ButPlus_Click(sender As Object, e As EventArgs) Handles ButPlus.Click
     If Label3.Text <> "" Then Exit Sub 'don't do anything if the sign is already there
     If DisplayNotEmpty() = False Then 'check there is number in the display
        Label1.Text = TextBox1.Text & " +" : Label4.Text = "+" 'sign in lavel4 is used to calculate
        FirstValue() 'sub call
     End If
End Sub
   'Minus Button
Private Sub ButMinus_Click(sender As Object, e As EventArgs) Handles ButMinus.Click
     If Label3.Text <> "" Then Exit Sub
     If DisplayNotEmpty() = False Then
        Label1.Text = TextBox1.Text & " -" : Label4.Text = "-"
        FirstValue()
     End If
End Sub
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    'Times Button
Private Sub ButTimes_Click(sender As Object, e As EventArgs) Handles ButTimes.Click
        If Label3.Text <> "" Then Exit Sub
        If DisplayNotEmpty() = False Then
            Label1.Text = TextBox1.Text & " ¸Y" : Label4.Text = "x"
            FirstValue()
        End If
End Sub
    'Divide Button
Private Sub ButDivide_Click(sender As Object, e As EventArgs) Handles ButDivide.Click
        If Label3.Text <> "" Then Exit Sub
        If DisplayNotEmpty() = False Then
            Label1.Text = TextBox1.Text & " /" : Label4.Text = "÷"
            FirstValue()
        End If
End Sub
    'percentage Button
    Private Sub ButPercent_Click(sender As Object, e As EventArgs) Handles ButPercent.Click
        If Label3.Text <> "" Then Exit Sub
        If DisplayNotEmpty() = False Then
            Label1.Text = TextBox1.Text & " % " : Label4.Text = "%"
            FirstValue()
        End If
End Sub
```

উপরের প্রতিটি সাবের কাজ যেহেতু একই ধরনের তাই একটির ওপর ব্যাখ্যা তুলে ধরাই যথেষ্ট হবে। সাবের প্রথম লাইন দ্বারা নিশ্চিত করা হচ্ছে ইতোমধ্যে ব্যবহারকারী কোনো সংকেতে ক্লিক করে নেন নি। যদি করে থাকেন তাহলে তো লেবেলে [Label3 -তে] সংকেত ঢুকানো হয়েই গেলো। সুতরাং [Exit Sub -দ্বারা] সাব থেকে বের হতে হবে।

দ্বিতীয় চেকটি হলো ডিসপ্লেতে সংখ্যা আছে কি না তা জানা। সংখ্যা থাকলেই [ফাংশনের ফলাফল] পরের লাইনে যেয়ে আমরা প্রথমে লেবেল১-এ ব্যবহারকারীকে কিছু তথ্য দিচ্ছি এবং দ্বিতীয়ত গোপন লেবেল৪-এ সংকেতটি সংরক্ষণ করছি। এরপর ফার্স্টভ্যালু নামক সাবরুটিন কল করেছি। যার কাজ হলো লেবেল৩-তে ডিসপ্লের সংখ্যা নিয়ে সংরক্ষণ করা ও ডিসপ্লে মুছে দেওয়া। এই কাজগুলো উপরের ৫টি সাব দ্বারা সম্পাদিত হয়। এবার আমরা ‘স্কোয়ার রুট’ [বর্গমূল] ক্যালকুলেশনের বিশেষ সাবটির ওপর তথ্যাদি তুলে ধরছি।

```
'Square Root Calculation and display sub
Private Sub ButSqr_Click(sender As Object, e As EventArgs) Handles ButSqr.Click
        If TextBox1.Text <> "" And Label1.Text = "" Then 'make sure number exists
            Dim SquareRoot As Double = Math.Sqrt(TextBox1.Text) 'get sqr into variable
            Label1.Text = TextBox1.Text & " Gi eM©g~j" 'bangla info display
            TextBox1.Text = Math.Round(SquareRoot, 8) 'put result into display
            Label2.Text = SquareRoot 'put result into memory
        End If
End Sub
```

বুঝতেই পারছেন, বর্গমূল বের করা অন্যান্য সকল কাজ থেকে ভিন্ন। এর কারণ হলো, বর্গমূলের জন্য একটি মাত্র সংখ্যার প্রয়োজন। অপরদিকে +, –, ÷, x ও % বের করতে দু’টি সংখ্যা থাকা জরুরী। সুতরাং ওগুলোর ক্যালকুলেশনও হবে একই ধরনের।

১৪তম পরিচ্ছেদ: ***Complete Application Tutorials***

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

উপরের সাবের প্রথম লাইনের কাজ হলো ডিসপ্লে ফিল্ডে একটি সংখ্যা থাকা নিশ্চিত করা। এই টেস্ট সত্যি ফলাফল রিটার্ন না করলে সাব থেকে বেরিয়ে যেতে হবে। যাক, টেস্ট সফল হওয়ার নিশ্চয়তা লাভের পর আমরা বর্গমূল বের করার কাজে যেতে পরি- যা পরবর্তী চারটি লাইনের ক্রিয়া। প্রথমেই আমরা একটি ভেরিয়েবল ডিক্লায়ার করে তাতে সংরক্ষণ করেছি সংখ্যাটির বর্গমূল। লক্ষ করুন, ভেরিয়েবল হবে Double Type -এটা মূলত দশমিক সংখ্যা টাইপ। এছাড়া বর্গমূল বের করতে আমরা ব্যবহার করেছি, সিস্টেম ফাংশন Math.Sqrt(Number) । এই ফাংশন একটি মাত্র ফ্লোটিং পয়েন্ট [দশমিক] সংখ্যা প্যারামিটার গ্রহণ করে। আমাদের ক্ষেত্রে তাহলো ডিসপ্লেতে লিখিত সংখ্যাটি। পরের লাইনের কোড দ্বারা লেবেল-১-এ বাংলায় কিছু তথ্য দেওয়া হয়েছে। লক্ষ করুন, প্রদর্শিত তথ্যাদি হলো টেক্সট বাক্সের সংখ্যা ও বাংলায় এ কথাটি: " এর বর্গমূল"। এর পরের লাইন দ্বারা ভেরিয়েবলে সংরক্ষিত বর্গমূল সংখ্যা [৮টি দশমিক স্থান পর্যন্ত] প্রদর্শন করেছি ডিসপ্লে ফিল্ডে। এখানেও একটি সিস্টেম ফাংশন কাজে লাগিয়েছি: Math.Round (SquareRoot, 8)। এই ফাংশন একটি ফ্লোটিং পয়েন্ট সংখ্যা ও আরেকটি ইন্টেজার সংখ্যা প্যারামিটার হিসাবে গ্রহণ করে। দ্বিতীয়টি দ্বারা ক'টি পর্যন্ত ডেসিমেল প্লেইসেজ থাকবে তা নির্দিষ্ট হয়। সাবরুটিনের সর্বশেষ লাইনের কাজ হচ্ছে, ফলাফল [বর্গমূল সংখ্যা] লুকানো লেবেল-২ এ মেমোরি হিসাবে সংরক্ষণ করা।

```
 'this sub clears last charater from the display
Private Sub ButClear_Click(sender As Object, e As EventArgs) Handles ButClear.Click
    If TextBox1.Text = "" Then Exit Sub 'make sure there is text to be deleted
    TextBox1.Text = Mid(TextBox1.Text, 1, (Len(TextBox1.Text) - 1)) 'collects all text except last one
    'and displays
End Sub
```

উপরের এ সাবটির কাজ হলো ডিসপ্লেতে প্রদর্শিত সর্বশেষ ডিজিটটি মুছে ফেলা। প্রথমে ডিসপ্লেতে সংখ্যা আছে তা নিশ্চিত করে এ কাজটি করা হয়েছে। লক্ষ করুন, Mid -হলো একটি সিস্টেম ফাংশন। এর দ্বারা নির্দিষ্ট পরিমাণ লেখা সংগ্রহ করা যায়। প্রথম (১) থেকে শেষেরটির আগেরটি পর্যন্ত আমরা সংগ্রহ করে পুনরায় ডিসপ্লেতে প্রদর্শন করছি এই ফাংশন ব্যবহার করে। Mid -ফাংশন বেশ ক'টি প্যারামিটার গ্রহণ করে। এখানে আমরা তিনটি প্যারামিটার দিয়েছি। প্রথমটি [TextBox1.Text] হলো ডসপ্লের পুরো লেখাটুকু। দ্বিতীয়টি হলে লেখার কোথায় থেকে সংগ্রহ শুরু হবে- সে ক্যারাক্টার পজিশন সংখ্যা [এখানে ১]। তৃতীয়টি হলো কী পরিমাণ লেখা সংগ্রহ করা হবে- তা নির্দিষ্টকরণ, এখানে শেষ ক্যারাক্টার পজিশনের আগেরটা পর্যন্ত। এভাবে করায় ব্যবহারকারী দেখবেন, শেষের সংখ্যাটি মুছে গেছে।

এবার আমরা ক্যালকুলেটর কোডের মূল সাবরুটিনটি নিয়ে বিস্তারিত আলোচনা করবো। এ সাবরুটিনটিই হলো পুরো প্রোগ্রামের মগজ। ইতোমধ্যে যা হয়েছে [উপরের সাব বাদে] তা ছিলো প্রস্তুতি গ্রহণ ক্রিয়া। একটি ক্যালকুলেটরে সর্বদাই সমান চিহ্ন [=] থাকবে। এ চিহ্নটি দ্বারাই কাঙ্ক্ষিত ফলাফল হিসাব করে ডিসপ্লেতে প্রদর্শিত হয়। সুতরাং আসুন, আমাদের বাংলা ক্যালকুলেটরের সমান চিহ্নের বোতামের ক্লিক ইভেন্ট সাবরুটিন কিভাবে কাজ করে তা তলিয়ে দেখি। প্রথমেই আমরা পুরে সাবরুটিনটি [এ পৃষ্ঠা ও পরের পৃষ্ঠায়] তুলে ধরছি।

```
 'calculate value and display
Private Sub ButEqual_Click(sender As Object, e As EventArgs) Handles ButEqual.Click
   If CheckValidity() = False Then Exit Sub 'check if all in order
   Dim CalValue As Double 'declare variable to hold result
   If Label4.Text = "+" Then 'check which calculation needed
        CalValue = Val(Label3.Text) + Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = Label1.Text & " " & TextBox1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display to 8 decimal places
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
   End If
```

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    If Label4.Text = "-" Then 'check which calculation needed
        CalValue = Val(Label3.Text) - Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = Label1.Text & " " & TextBox1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display to 8 decimal places
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
    If Label4.Text = "x" Then 'check which calculation needed
        CalValue = Val(Label3.Text) * Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = Label1.Text & " " & TextBox1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display to 8 decimal places
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
    If Label4.Text = "÷" Then 'check which calculation needed
        'check if user trying to divide by 0
        If Val(TextBox1.Text) = 0 Then
            MsgBox("আহ! ০-দ্বারা ভাগ করা তো বৈধ নয়!") 'if so show a message and exit
            Exit Sub
        End If
        CalValue = Val(Label3.Text) / Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = Label1.Text & " " & TextBox1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
    If Label4.Text = "%" Then 'check which calculation needed
        CalValue = (Val(Label3.Text) / 100) * Val(TextBox1.Text) 'actual calculation into variable
        Label1.Text = TextBox1.Text & " Gi " & Label1.Text 'Info display
        TextBox1.Text = Math.Round(CalValue, 8) 'display to 8 decimal places
        Label3.Text = "" 'clear hidden value
        Label2.Text = CalValue 'result into memory
        Exit Sub
    End If
End Sub
```

প্রথমেই মনে করিয়ে দিচ্ছি, এ সাবরুটিন হলো সমান চিহ্নের বোতামের ক্লিক ইভেন্ট। প্রথমেই সবকিছু ঠিকমতো ডিসপ্লেতে ঢুকানো হয়েছে কি না তা দেখতে হবে। এ কাজের জন্য একটি ফাংশন বানিয়েছি- যার বর্ণনা শেষে আসবে। এ ফাংশনের কাজ হলো 'সত্য' কিংবা 'মিথ্যা' ফলাফল রিটার্ন করা। আমাদের উদ্দেশ্য সত্য ফলাফল পাওয়া। না পেলে সাবরুটিন থেকে বের হয়ে যেতে হবে। ফলাফল সত্য হলে পরের লাইনে যেয়ে একটি ডাবল টাইপ ভেরিয়েবল [CalValue] ঘোষণা করেছি যাতে থাকবে ফলাফল সংখ্যাটি। পরের লাইনে যেয়ে চেক করা হয়েছে কোন্ ক্যালকুলেন হবে [+ যোগ হবে কি?]। কথা সত্য হলে, এ সময় অবশ্যই লুকানো লেবেল৪-এ '+' লিখা থাকবে। ধরে নিন + লিখা আছে। এবার যোগ অঙ্ক কষতে হবে। পরের লাইনে আগে ঘোষিত ভেরিয়বলে সংখ্যা দু'টি যোগ করে সংরক্ষণ করা হয়েছে। এরপর লেবেল১-এ কিছু দৃশ্যমান তথ্য দেওয়া হয়েছে, ব্যবহারকারী কোন্ সংখ্যার সঙ্গে কোন্ সংখ্যা যোগ দিলেন তা বুঝানের জন্য। পরের লাইন দ্বারা যোগফলটি ৮টি ডেসিমেল প্লেইস পর্যন্ত প্রদর্শনের জন পরিবর্তন করে ডিসপ্লে ফিল্ডে দেখানো হয়েছে। পরের দু'টি লাইনের প্রথমটি দ্বারা লুকানো লেবেল-৩ এর মান মুছে ফেলার পর দ্বিতীয় লাইন দ্বারা লুকানো লেবেল২-এ প্রাপ্ত মানটি মেমোরি হিসাবে সংরক্ষিত হবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

সাবরুটিনের পরবর্তী ৪টি কোড ব্লকই মূলত একই ধরনের। এদের স্ট্রাকচার ও ফাংশন প্রায় একই। শুধুমাত্র –, x, ÷ ও % ক্যালকুলেশন লাইনগুলো একটু ভিন্ন- এই যা। সুতরাং সবকিছু পুনরায় বলার মানে নেই। তবে সামান্য কিছুটা পার্থক্য বিদ্যমান- যার কারণে কিছুটা ব্যাখ্যাও প্রয়োজন।

(ক) প্রথমত, ÷ অঙ্কের ক্ষেত্রে একটি ব্যাপার সকলেরই জানা- কোনো সংখ্যাকে ০ [শূন্য] দ্বারা ভাগ করা যায় না। এটা গাণিতিকভাবে অসম্ভব। সুতরাং কেউ যদি ০ দ্বারা ভাগ করার চেষ্টা করেন, তখন প্রোগ্রাম যাতে একটি মেসেজ প্রদর্শন করে সাবরুটিন থেকে বেরিয়ে পড়ে সে ব্যবস্থা করতে হবে। আমরা বাংলায় একটি মেসেজ প্রদর্শন করার ব্যবস্থা করেছি।

(খ) দ্বিতীয়ত, সবগুলো ক্যালকুলেশনের লাইনে ভ্যাল [VAL] নামক একটি ফাংশন ব্যবহার করেছি- লক্ষ করেছেন তো? এটার কাজ হলো ব্র্যাকেটের ভেতরস্থ স্ট্রিংটিকে সংখ্যায় পরিণত করা। এটা অতিরিক্ত সতর্কতা হেতু ব্যবহার করেছি। বাস্তবে অধিকাংশ ক্ষেত্রে ভিজুয়্যাল বেসিক এরূপ স্ট্রিংকে এমনিতেই সংখ্যায় পরিণত করে থাকে। তবে ক্যালকুলেটরের ক্ষেত্রে তো নিশ্চিতভাবে সংখ্যা বানানো ছাড়া বিকল্প নেই- তা-ই এ ব্যবস্থা।

ক্যালকুলেটরের কোডের ব্যাখ্যা প্রায় শেষ। দু'টি মাত্র কোড ব্লক বাকী আছে। এর প্রথমটি হলো ফাংশন ও দ্বিতীয়টি সাবরুটিন। প্রথমটি জরুরী আর দ্বিতীয়টি ব্যক্তিগত। চাইলে এটা দিতে পারেন- না দিলে অসুবিধে নেই। অবশ্য প্রোগ্রামার হিসাবে নাম ও তথ্যাদি তো থাকবে আপনারই, আমার নয়!

```
    'Function returns true if the conditions for calculation is fulfilled, otherwise returns false
Function CheckValidity()
        If TextBox1.Text <> "" And Label1.Text <> "" And Label3.Text <> "" And Label4.Text <> "' Then" Then
            Return True
        End If
        Return False
End Function
'This sub is to show some info about the programmer- Button1 is 'About the Programmer' button at the top
 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        MsgBox("Engineer Azizul Bari" & vbCrLf & "Diploma in Science & Technology, England" & vbCrLf & _
        "Khanqa-E-Aminia-Asgaria Research Section" & vbCrLf & "Ali Centre, Subidbazar, Sylhet, Bangladesh" & _
          vbCrLf & "engineerazizulbari@gamil.com")
End Sub
```

ফাংশনের ফলাফল সত্যি হলে ক্যালকুলেশন হবে। এটা সত্যি হওয়ার শর্তগুলো হলো: ডিসপ্লে, প্রদর্শন লেবেল১, লুকানো লেবেল৩ এবং লুকানো লেবেল৪ খালি না হওয়া। এ সবগুলোর মধ্যে তথ্য থাকলেই বুঝা যাবে ক্যালকুলেশন হবে।

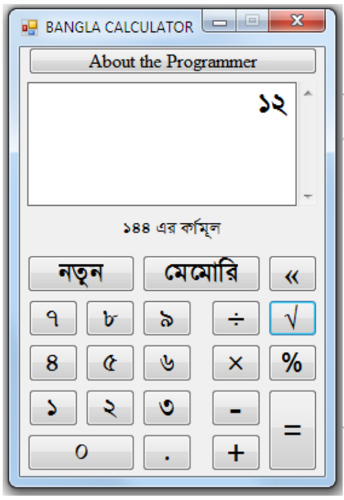
সবশেষে উপরোক্ত শেষের সাবরুটিনে দীর্ঘ লাইন কোড ফিল্ডে লিখার একটি উপায় দেখানো হয়েছে। সেসাথে একটি কোড দেওয়া হয়েছে যার কাজ ইনফিড বা রিটার্ন কিস্ট্রোকের ফাংশন আঞ্জাম দেওয়া। যথাক্রমে প্রথমটি হলো আন্ডারস্কোর [_] আর দ্বিতীয়টি vbCrLf কিওয়ার্ড।

বাকী থাকলো আপনার সৃষ্টি বাংলা ক্যালকুলেটর টেস্ট করে ডিস্ট্রিবিউট করা। পরের পৃষ্ঠায় এ সম্পর্কে কিছু তথ্য ও ব্যাখ্যা প্রদান করে এ টিউটোরিয়ালের ইতি টানবো।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

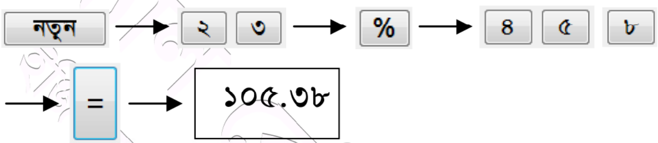
পরীক্ষণ ও বিস্তারণ: বাংলা ক্যালকুলেটর এ্যাপলিকেশনের প্রোগ্রামিং বা কোডিং পর্যায় শেষ হলো। এবার একে পরীক্ষা করতে হবে। এজন্য প্রয়োজন বিল্ড করে পরিচালনা। তাই এক্ষুণি পরীক্ষা শুরু হয়ে যাক। নিচের চিত্রে চলাকালীন জানালাটি দেখতে পাচ্ছেন।

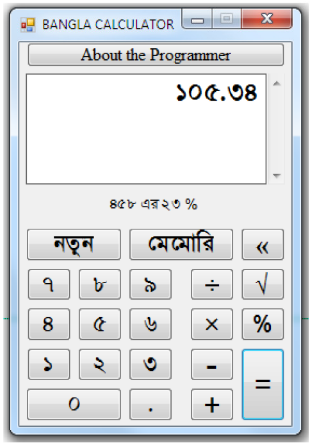


আমরা মূল ডিসপ্লে ফিল্ডে সংখ্যা ১২ দেখতে পাচ্ছি। ডিসপ্লের নিচে আছে '১৪৪ এর বর্গমূল' কথাটি। সুতরাং বুঝাই যাচ্ছে, ১৪৪ এর বর্গমূল হলো ১২- এই ক্যালকুলেন করেছি। এটা করতে প্রথমে ক্যালকুলেটরে আপনি যেভাবে সংখ্যা ঢুকাবেন তাহলো:



অনুরূপ ৪৫৮ -এর ২৩ শতাংশ বের করতে আপনি যে ভাবে সংখ্যা ঢুকাবেন তাহলো (ডানের চিত্র):





এ দু'টো ফাংশনের কাজ দেখানোর কারণ হলো, এগুলোই আমাদের বাংলা ক্যালকুলেটরের বিষেশ ফাংশন। বাকি চারটে পাটিগণিতের ফাংশন [+, – ÷ ও x] সচরাচর ব্যবহৃত ক্যালকুলেটরের মতোই কাজ করে। সুতরাং ওগুলোর অপারেশন এখানে দেখাচ্ছি না। তবে আপনার উচিৎ হবে সবগুলো ফাংশনই একে এক পরীক্ষা করা। আপনার এ সৃষ্টিকে তো শেষ পর্যন্ত গ্রাহকদের নিকট পৌঁছুনোর দরকার! আর এ কাজের নামই বিস্তারণ।

ভিজুয়্যাল স্টুডিও এক্সপ্রেস ফর ডেস্কটপ পূর্ণাঙ্গ এ্যাপলিকেশন বিস্তারণে আমাদেরকে বিষেশ সহযোগিতা করে। টেস্টিং স্তরের কাজ সম্পন্ন হওয়ার পর আপনি সহজেই কয়েকটির পদক্ষেপে সফটওয়্যার ডিস্ট্রিবিউটের জন্য ইনস্টলার তৈরি করতে পারেন। আমরা গ্রন্থের শেষে ইনস্টলার তৈরির পদ্ধতির ওপর বিস্তারিত আলোচনা করবো। এবার আসুন একটি উন্নতমানের বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার তৈরি করি! এ প্রোগ্রামটি দীর্ঘ, সময়সাক্ষেপক এবং কিছুটা জটিল। তবে প্রোগ্রাম করা শেষ হলে আপনি অবশ্যই আনন্দিত হবেন। কবি বলেন নি, এ সৃষ্টি সুখের উল্লাসে!

Application 3 : Bangla / English Unit Conversion Software - বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার।

সফটওয়্যার তৈরির লক্ষ্য: আমরা একটি বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার তৈরি করবো যাতেকরে ব্যবহারকারী অতি সহজে বিভিন্ন এককের মধ্যে সম্পর্ক, পরিবর্তন, সাংখ্যিক ফলাফল ইত্যাদি বের করতে সক্ষম হন।

ব্যাপ্তি: বিভিন্ন এককের মধ্যে শত শত সম্পর্ক বিদ্যমান। তবে আমাদের টার্গেট হলো সচরাচর ব্যবহৃত এককগুলো নিয়ে কাজ করা। সেমতে আমরা নিম্নের [পরের পৃষ্ঠার] টেবিলে দেওয়া ইউনিটগুলো নিয়ে কাজ করবো।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

দূরত্ব (Length)			দূরত্ব (Length)			দূরত্ব (Length)		
নাম	সম্পর্ক	নাম	নাম	সম্পর্ক	নাম	নাম	সম্পর্ক	নাম
মাইল	১=১.৬০৯৩	কি.মি	কি.মি	১=১০৯৩.৬১৩৩	গজ	ইঞ্চি	১=১/৬৩৩৬০	মাইল
মাইল	১=১৭৬০	গজ	কি.মি	১=১/১.৬০৯৩	মাইল	ইঞ্চি	১=১/৩৯৩৭০.০৭৮৮	কি.মি
মাইল	১=৫২৮০	ফুট	কি.মি	১=১০০০	মিটার	ইঞ্চি	১=১/৩৯.৩৭০০৭৮৮	মিটার
মাইল	১=১৬০৯.৩৪৪	মিটার	কি.মি	১=৩৯৩৭০.০৭৮৮	ইঞ্চি	ইঞ্চি	১=২৫.৪	মিমি
মাইল	১=১৬০৯৩৪৪	মিমি	কি.মি	১=৩২৮০.৮৩৯৯	ফুট	ইঞ্চি	১=১/৩৬	গজ
মাইল	১=৬৩৩৬০	ইঞ্চি	কি.মি	১=১০০০০০০	মিমি	ইঞ্চি	১=১/১২	ফুট
গজ	১=১/১৭৬০	মাইল	মিটার	১=১০০০	মিমি	মিমি	১=১/২৫.৪	ইঞ্চি
গজ	১=১/১০৯৩.৬১৩৩	কি.মি	মিটার	১=১/১০০০	কি.মি	মিমি	১=১/১০০০	মিটার
গজ	১=৩	ফুট	মিটার	১=৩.২৮০৮৪	ফুট	মিমি	১=১/৯১৪.৪	গজ
গজ	১=১/১.০৯৩৬	মিটার	মিটার	১=৩৯.৩৭০০৭৮৮	ইঞ্চি	মিমি	১=১/১৬০৯৩৪৪	মাইল
গজ	১=৯১৪.৪	মিমি	মিটার	১=১/১৬০৯.৩	মাইল	মিমি	১=১/১০০০০০০	কি.মি
গজ	১=৩৬	ইঞ্চি	মিটার	১=১.০৯৩৬	গজ	মিমি	১=১/৩০৪.৮	ফুট
ফুট	১=১/৫২৮০	মাইল	ফুট	১=১/৩.২৮০৮৪	মিটার	দূরত্ব একক রূপান্তরের জন্য মোট ৪২টি ফাংশন ব্যবহার করবো। ভাববেন না, বাস্তবে এ কাজ অতি সহজ।		
ফুট	১=১/৩২৮০.৮৩৯৯	কি.মি.	ফুট	১=৩০৪.৮	মিমি			
ফুট	১=১/৩	গজ	ফুট	১=১২	ইঞ্চি			

ক্ষেত্র (Area)			ক্ষেত্র (Area)		
নাম	সম্পর্ক	নাম	নাম	সম্পর্ক	নাম
বর্গমাইল	১=২.৫৯	বর্গকি.মি	বর্গমিটার	১=১০০০০০০	বর্গমিমি
বর্গমাইল	১=১৮৬৮৩৪২৮.১১	বর্গমিটার	বর্গমিটার	১=১/১৮৬৮৩৪২৮.১১	বর্গমাইল
বর্গমাইল	১=২৭৮৭৮৪০০	বর্গফুট	বর্গমিটার	১=১/১০০০০০০	বর্গকি.মি
বর্গমাইল	১=৩০৯৭৬০০	বর্গগজ	বর্গমিটার	১=১০.৭৬৩৯১০৪১৬৭	বর্গফুট
বর্গমাইল	১=৪০১৪৫০৫৬৫৮.০০৬৬	বর্গইঞ্চি	বর্গমিটার	১=১.১৯৫৯৯	বর্গগজ
বর্গমাইল	১=২৫৮৯৯৮৮১১০৩৩৬	বর্গমিমি	বর্গমিটার	১=১৫৫০.০০৩১	বর্গইঞ্চি
বর্গকি.মি	১=১/২.৫৯	বর্গমাইল	বর্গফুট	১=১/১০.৭৬৩৯১০৪১৬৭	বর্গমিটার
বর্গকি.মি	১=১৫৫০০০৩১০০.০০৬২	বর্গইঞ্চি	বর্গফুট	১=১/২৭৮৭৮৪০০	বর্গমাইল
বর্গকি.মি	১=১০০০০০০	বর্গমিটার	বর্গফুট	১=১/৯	বর্গগজ
বর্গকি.মি	১=১১৯৫৯৯০.০৪৬৩	বর্গগজ	বর্গফুট	১=১৪৪	বর্গইঞ্চি
বর্গকি.মি	১=১০০০০০০০০০০০০	বর্গমিমি	বর্গফুট	১=৯২৯০৩.০৪	বর্গমিমি
বর্গকি.মি	১=১০৭৬৩৯১০.৪১৬৭	বর্গফুট	বর্গফুট	১=১/১০৭৬৩৯১০.৪১৬৭	বর্গকি.মি
বর্গগজ	১=১/৩০৯৭৬০০	বর্গমাইল	বর্গইঞ্চি	১=১/৪০১৪৫০৫৬৫৮.০০৬৬	বর্গমাইল
বর্গগজ	১=১/১১৯৫৯৯০.০৪৬৩	বর্গকি.মি	বর্গইঞ্চি	১=১/১৫৫০০০৩১০০.০০৬২	বর্গকি.মি
বর্গগজ	১=৮৩৬১২৭.৩৬	বর্গমিমি	বর্গইঞ্চি	১=১/১২৯৬	বর্গগজ
বর্গগজ	১=১/১.১৯৫৯৯	বর্গমিটার	বর্গইঞ্চি	১=১/১৪৪	বর্গফুট
বর্গগজ	১=১২৯৬	বর্গইঞ্চি	বর্গইঞ্চি	১=১/১৫৫০.০০৩১	বর্গমিটার
বর্গগজ	১=৯	বর্গফুট	বর্গইঞ্চি	১=৬৪৫.১৬	বর্গমিমি
বর্গমিমি	১=১/২৫৮৯৯৮৮১১০৩৩৬	বর্গমাইল	ওজন (Weight)		
বর্গমিমি	১=১/৮৩৬১২৭.৩৬	বর্গগজ	কি.গ্রা	১=১/০.৯৩৩১০	সের
বর্গমিমি	১=১/৬৪৫.১৬	বর্গইঞ্চি	কি.গ্রা	১=১০০০	গ্রাম
বর্গমিমি	১=১/১০০০০০০	বর্গমিটার	কি.গ্রা.	১=১০০০/১১.৬৬	ভরি
বর্গমিমি	১=১/১০০০০০০০০০০০০	বর্গকি.মি	গ্রাম	১=১/৯৩৩.১০	সের
বর্গমিমি	১=১/৯২৯০৩.০৪	বর্গফুট	গ্রাম	১=১/১০০০	কি.গ্রা
দূরত্বে আছে ৭টি, ক্ষেত্রতে আছে ৭টি, তাপে আছে ৩টি, ওজনে আছে ৪টি এবং সার্ভেতে আছে ৩টি একক। সুতরাং মোট ২৪টি একক নিয়ে আমাদের কাজ।			গ্রাম	১=১/১১.৬৬	ভরি
			ভরি	১=১১.৬৬	গ্রাম

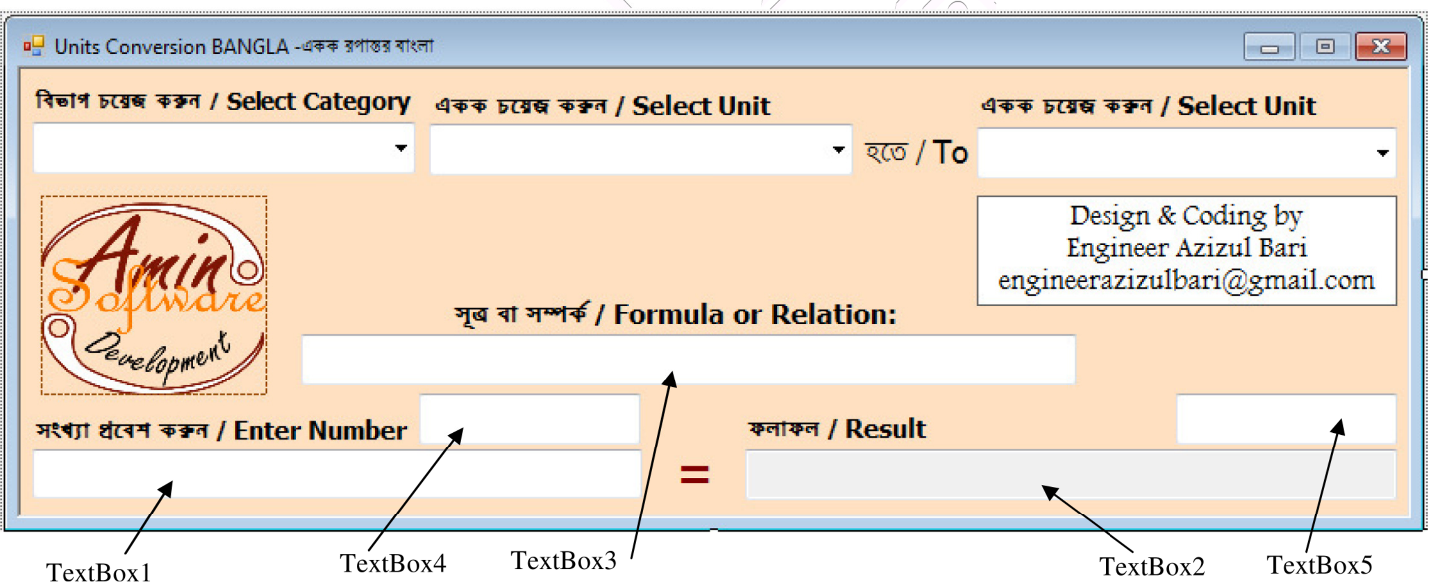
১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

তাপ (Temperature)					
ডিগ্রী সেলসিয়াস	°F = (°C × 180/100) + 32)	ডিগ্রী ফারেনহাইট	ভরি	১=১/৮৫.৭৬৩২৯৩৩১	কি.গ্রা
ডিগ্রী ফারেনহাইট	°C = (°F - 32) × 100/180	ডিগ্রী সেলসিয়াস	ভরি	১=১/৮০.০১৭১৫	সের
কেলভিন	°C = (K - 273.15)	ডিগ্রী সেলসিয়াস	সের	১=০.৯৩৩	কি.গ্রা
ডিগ্রী সেলসিয়াস	K = (°C + 273.15)	কেলভিন	সের	১=৯৩৩.১	গ্রাম
কেলভিন	°F = (K × 9/5) – 459.67	ডিগ্রী ফারেনহাইট	সের	১=৮০.০১৭১৫	ভরি
ডিগ্রী ফারেনহাইট	K = (°F - 32) × 5 / 9 + 273.15	কেলভিন			
জমির হিসাব (Survey)					
বর্গফুট	১=১/৪৩৫৬০.১৭৪২	একর	একর	১=৪৩৫৬০.১৭৪২	বর্গফুট
বর্গফুট	১=১/৪৩৫.৬০১৭৪২	শতাংশ	শতাংশ	১=৪৩৫.৬০১৭৪২	বর্গফুট
একর	১=১০০	শতাংশ	শতাংশ	১=১/১০০	একর

উপরের টেবিলগুলোতে আমাদের ২৪টি এককের একটা আরেকটার সাথে সম্পর্ক কী তা দেখিয়েছি। এবার আমরা একক পরিবর্তন প্রোগ্রামের ইউজার ইন্টারফেইস ডিজাইনের কাজে লেগে যেতে পারি। আমাদের উদ্দেশ্য হলো ব্যবহারকারী প্রথমে চয়েজ করবেন ক্যাটাগরি। এরপর এ চয়েজকৃত ক্যাটাগরির কোন্ একক থেকে অপরটিতে পরিবর্তন করবের তা চয়েজ করবেন। সবশেষে চয়েজ করবেন কোন্ ইউনিটে পরিবর্তন হবে। অবশ্য এক পর্যায়ে যে একক থেকে পরিবর্তন করবেন তার মানও [সংখ্যা] তাকে একটি ফিল্ডে প্রবেশ করতে হবে।

অপর যে ব্যাপারটি তাহলো প্রোগ্রামটি নিরবচ্ছিন্নভাবে রান করবে। এর অর্থ হলো একক চয়েজ কিংবা মান ঢুকানোর পরই ফলাফল এসে যাবে। এজন্য আলাদাভাবে কোনো বোতামে ক্লিক করার দরকার পড়বে না। যাক ফর্মের একটি ডিজাইন নিচের চিত্রে তুলে ধরেছি। একটু দেখে নিন। এরপর বর্ণিত পদক্ষেপগুলো অনুসরণ করে নিজেই প্রজেক্ট তৈরি করে ফর্মটি ডিজাইন করুন।



১. একটি নতুন উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করুন। নামকরণ করুন: Bangla - English Units Conversion।

২. ফর্মের সাইজ = ৪৮১, ৩০৬; ফর্মের Text = একক রূপান্তর বাংলা - ইংরেজি। বাংলা কথাগুলো ইউনিকোডে লিখবেন [বিজয় কিবোর্ডে ইউনিকোডে লিখার ফাংশন আছে]। ফর্মের ফন্ট = Tahoma, 12pt ; ফর্মের MaximizeBox = False; ফর্মের FormBorderStyle = Fixed3D; ফর্মের BackColor = 255,224,192। ব্যস! আর সকল প্রপার্টি অপরিবর্তিত থাকবে।

৩. ফর্মে যোগ করুন তিনটি কম্বোবক্স - ComboBox1, ComboBox2 এবং ComboBox3।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

৪. ফর্মে যোগ করুন ৭টি লেবেল ও ৫টি টেক্সট্ বাক্স। কোন্ লেবেলে কী লিখা থাকবে তা চিত্রে দেওয়া আছে। তবে ৫টি টেক্সট্ বাক্সের কোন্‌টি কোথায় হবে তা চিত্রে নির্দেশ মুতাবিক স্থাপন করুন। লেবেলের সকল লেখা মূলত তথ্য প্রদানের জন্য। আপনি লেবেলের Text প্রোপার্টিতে প্রথমে বাংলায় [ইউনিকোডে লিখবেন] এরপর ইংরেজিতে লিখবেন।
৫. টেক্সট্ বাক্স ১ ও ২ এর ফন্ট প্রেপার্টি = TonnyBanglaMJ, 14pt অথবা SutonnyMJ, 14pt। শুধুমাত্র টেক্সট্ বাক্স ২ এর ForeColor = Maroon এবং ReadOnly = True হবে। আমরা চাই এ বাক্সে ফলাফল প্রদর্শিত হবে। সুতরাং ব্যবহারকারী এতে যাতে কিছু লিখতে পারেন না- সেটা নিশ্চিত করবে শেষোক্ত প্রপার্টি।

ফর্মে আরো দু'টি বস্তু আছে: একটি টেক্সড্ বাক্স ও একটি পিকচার বক্স। এগুলো যুক্ত করা জরুরী নয়। কিসের জন্য এগুলো দিয়েছি তা-তো বুঝতেই পারছেন! তবে আপনি নিজেরটা দেবেন- এটা আশা করা যায়- কেমন?

যাক, এই তো! আমাদের প্রোগ্রামের ইউজার ইন্টারফেইস ডিজাইন হয়ে গেলো। একবার Start মেনু চয়েজ করে প্রোগ্রাম রান করে দেখুন। সব ঠিক আছে তো? আসুন তাহলে, মূল কাজে লেগে যাই- প্রোগ্রাম কোডিং!

কোডিং: ব্যক্তিগতভাবে আমি সহজ-সরল এ্যালগোরিথম ও লজিক ব্যবহারে বিশ্বাসী। এর কারণে যদি কোডিং দীর্ঘই হয় তাতে কোনো অসুবিধা নেই। প্রোগ্রাম বেঁটে করতে যেয়ে খুব জটিল ফাংশন, সাবরুটিন, ক্লাস, মডিউল ইত্যাদি তৈরি করার পক্ষপাতি আমি নই। হ্যাঁ, প্রয়োজনে এগুলোও করতে হবে- তবে সরলতা অনুসরণ করে। এভাবে প্রোগ্রাম করলে পরবর্তীতে ডিবাগিং অনেকটা সহজ হয়। যাক, আমাদের এ প্রোগ্রামটি বাস্তবে খুব একটা সহজ নয়। প্রথম কাজ হলো সঠিকভাবে বিভিন্ন কম্বো বক্স ও ফিল্ডে তথ্য প্রদর্শন করতে হবে। এজন্য প্রথমেই আমরা ডিফোল্ট জানালা প্রদর্শনের ব্যবস্থা করেছি। এরপর ব্যবহারকারীর চয়েজ অনুযায়ী প্রদর্শন ও ক্যালকুলেশন করতে হবে। তাহলে আসুন, কোডগুলো ধীরে ধীরে দেখে নিই ও প্রত্যেকটি সম্পর্কে ব্যাখ্যা তুলে ধরি।

আমরা তিনটি আলাদা এডিটর ফিল্ডে পূর্ণ কোড লিখেছি- আসলে লিখতে হয়েছে। এগুলো হলো:
১. ফর্মের কোড [Form1.vb*] ।
২. প্রদর্শনের জন্য একটি ফাংশন মডিউল [AllDisplayFunctionsModule.vb] ।
৩. সকল ক্যালকুলেশনের জন্য একটি ফাংশন মডিউল [CalculationModule.vb] ।

১. ফর্মের কোড [Form1.vb*]: এ কোডগুলোই প্রোগ্রাম চলাকালে সর্বপ্রথম রান করে। ফর্মের Public Class Form1 নামক সাবরুটিনের ভেতরই যাবতীয় কোড থাকতে হবে। এ সাবরুটিন থেকেই প্রয়োজনে আমরা বাকি দু'টো মডিউলের ফাংশনগুলো কল করতে হবে। সুতরাং ফর্মের ক্লাস সাবরুটিনটি কিভাবে কাজ করে তা জানা আমাদের জন্য একান্ত জরুরী। এর ভেতর আছে ১৮টি সাবরুটিন। এর মধ্যে ৭টি সিস্টেম ইভেন্ড সাবরুটিন এবং বাকি ১১টি আমাদের কর্তৃক ডিক্লায়রকৃত সাবরুটিন। অর্থাৎ এগুলো আমরা ইভেন্ট সাবরুটিনের ভেতর থেকে কল করে করে ব্যবহার করেছি বিভিন্ন কাজ সম্পাদন করতে। এ পর্যায়ে আমরা সবগুলো সাবরুটিনের নাম তালিকাভুক্ত করার প্রয়াস পাচ্ছি। নামের পাশে সাবরুটিনের কাজ কি তা-ও সংক্ষিপ্তভাবে বুঝিয়ে দিয়েছি।
১. সিস্টেম ইভেন্ট সাব: Form1_Load -এটা প্রোগ্রাম চলাকালে ফর্ম লোডের সময় রান করে।
২-৪. সিস্টেম ইভেন্ট সাব (২, ৩ ও ৪): ComboBox[1,2,3]_SelectedIndexChanged -এটা কম্বো বাক্সের ইনডেক্স সিলেকশনের ইভেন্ট সাব। আমরা এটা ধরে নিজেদের চাহিদানুযায়ী কাজ করেছি।
৫. সিস্টেম ইভেন্ট সাব: TextBox1_KeyPress -এটা হলো টেক্সড্ বাক্সে কিবোর্ড থেকে কিছু লিখার উভেন্ট সাব। আমার এটাকে ধরে শুধুমাত্র সংখ্যা ছাড়া যাতে আর কিছু ব্যবহারকারী কিবোর্ড থেকে প্রবেশ করতে পারেন না- সেটা নিশ্চিত করেছি। এভাবে নিয়ন্ত্রণ না করলে প্রোগ্রাম ক্রাশ করবে।
৬. সিস্টেম ইভেন্ট সাব: TextBox1_TextChanged -এটাও টেক্সট্ বাক্সের একটি ইভেন্ট যাকে আমরা ধরে কিছু ক্রিয়া সম্পাদন করেছি।

১৪তম পরিচ্ছেদ: ***Complete Application Tutorials***

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

৭. আমাদের সৃষ্ট সাব: MileMileKilo() -এটা ফর্মের লোড [Form1_Load] সাব থেকে কল করা হয়েছে।
৮. আমাদের সৃষ্ট সাব: Combo1One() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
৯. আমাদের সৃষ্ট সাব: AreaMileKilo() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
১০. আমাদের সৃষ্ট সাব: TempDegFahren() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
১১. আমাদের সৃষ্ট সাব: WeightKiloSher() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
১২. আমাদের সৃষ্ট সাব: SurveyFunction() -এটা সিস্টেম ইভেন্ট সাব ComboBox1_SelectedIndexChanged - থেকে কল করা হয়েছে।
১৩. আমাদের সৃষ্ট সাব: CalculateAndDisplayLength() -এটার কাজ হলো 'দূরত্ব' ক্যাটাগরির কোন্ একক থেকে কোন্ একক পরিবর্তন হবে তা সনাক্ত করে মডিউলের ফাংশন কল করা। এরপর ফাংশন থেকে প্রাপ্ত ফলাফল ডিসপ্লে ফিল্ড টেক্সট্ বাক্স ২-এ প্রদর্শন করা। এ সাবটি উপরের (১৩ নং) সিস্টেম সাব থেকে কল করা হয়।
১৪. আমাদের সৃষ্ট সাব: CalculateAndDisplayArea() -এটাও উপরোক্ত সাবের মতো কাজ করে। পার্থক্য হলো, এর কাজ 'ক্ষেত্র' ক্যাটাগরির সঙ্গে সম্পৃক্ত।
১৫. আমাদের সৃষ্ট সাব: CalculateAndDisplayTemp() -এটাও উপরোক্ত সাবের মতো কাজ করে। পার্থক্য হলো, এর কাজ 'তাপ' ক্যাটাগরির সঙ্গে সম্পৃক্ত।
১৬. আমাদের সৃষ্ট সাব: CalculateAndDisplayWeight() -এটাও উপরোক্ত সাবের মতো কাজ করে। পার্থক্য হলো, এর কাজ 'ওজন' ক্যাটাগরির সঙ্গে সম্পৃক্ত।
১৭. আমাদের সৃষ্ট সাব: CalculateAndDisplaySurvey() -এটাও উপরোক্ত সাবের মতো কাজ করে। পার্থক্য হলো, এর কাজ 'সার্ভে' ক্যাটাগরির সঙ্গে সম্পৃক্ত।

উপরোক্ত ১৭টি সাবরুটিনের সবই প্রোগ্রামের মেইন ক্লাস সাবরুটিন তথা Public Class Form1 -এর ভেতর অবস্থান করে। আগেই বলেছি দু'টি মডিউল আমরা নিজেই ডিফাইন করেছি আলাদাভাবে। এগুলোর ভেতর বেশ কিছু সাব ও ফাংশন আছে। মডিউল দু'টোর ব্যাপ্তি 'পাবলিক' হওয়ায় আমরা এসব সাব ও ফাংশন ফর্মের ক্লাস থেকে কল করার সুযোগ পেয়েছি। এ পর্যায়ে মডিউলে ডিফাইন করা প্রতিটি সাব সম্পর্কে বিস্তারিত বলার প্রয়োজন মনে করছি না। পরের পৃষ্ঠাগুলোতে আমরা সকল কোড একে একে তুলে ধরেছি। প্রথমে এসেছে Public Class Form1 -এর ভেতরস্থ সকল কোড। বিস্তারিত ব্যাখ্যা শেষে যথাক্রমে Public Module AllDisplayFunctionsModule -এর ভেতরস্থ কোড ও Public Module CalculationModule -এর ভেতরস্থ সকল কোড তুলে ধরেছি। প্রত্যেকটির ওপর বিস্তারিত ব্যাখ্যা তো অবশ্যই আছে। শিক্ষার্থীদের ধৈর্যধারণ ক্ষমতা আরো বেশি থাকা চাই। পুরো কোড তুলে ধরতে বইয়ের ১৫ পৃষ্ঠা লেগেছে। কোডের প্রতিটি লাইন ব্যাখ্যার আলোকে পাঠ করে বুঝতে হবে। সুতরাং আসুন, এবার পুরো কোড দেখে নিই ও বুঝার চেষ্টা করি। এই কোডটি কিভাবে কাজ করে তা যদি আমরা সঠিকভাবে বুঝতে সক্ষম হই তাহলে ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং সম্পর্কে তো বেশ দূরে অগ্রসর হয়ে গেলাম! তাই না? অবশ্য পরের প্রোগ্রামটি আরো একটু কঠিন! সুতরাং ধৈর্যের মাত্রা তখন আরো বাড়াতে হবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

১. ফর্মের কোড [Form1.vb*]

```
Public Class Form1
    'Global Variable Declarations
    'CatItem -Combobox1 Index, FromItem -Combobox2 Index, ToItem -Combobox3 Index
    Dim CatItem As Integer : Dim FromItem As Integer : Dim ToItem As Integer
    Dim ComBox2SelItemIdex As Integer 'to keep selected item from combo [combo2]
    Dim ComBox3SelItemIndex As Integer 'to keep selected item from combo [combo3]
    Dim EnteredNumber As Double 'to keep number item entered by the user
    Dim Combo1List(4) As String : Dim ComboInx0Combo2List(6) As String
    Dim ComboInx1Combo2List(6) As String
    Dim ComboboxInx2Combo2List(2) As String : Dim ComboboxInx3Combo2List(3) As String : Dim ComboboxInx4Combo2List(2)
As String
    '==================================================================================
    'This Sub Initialises the default Text and Values. All changes during calculations.
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        'Declare array variables for ComboBoxListItems and populate them.
        Combo1List = {"দূরত্ব / Length", "ক্ষেত্র / Area", "তাপ / Temperature", "ওজন / Weight", "জমির হিসাব / Survey"}
        ComboInx0Combo2List = {"মাইল / Mile (mi)", "কিলোমিটার / Kilometre (km)", "মিটার / Metre (m)", _
                    "মিলিমিটার / Millimetre (mm)", "গজ / Yard (yd)", "ফুট / Foot (ft)", "ইঞ্চি / Inch (in)"}
        ComboInx1Combo2List = {"বর্গমাইল / Sq. mi", "বর্গকিলোমিটার / Sq. Km", "বর্গমিটার / Sq. mt", "বর্গমিলিমিটার / Sq. mm", _
                    "বর্গগজ / Sq. yd", "বর্গফুট / Sq. ft", "বর্গইঞ্চি / Sq. in"}
        ComboboxInx2Combo2List = {"ডিগ্রী.সেলসিয়াস / °C", "ডিগ্রী.ফারেনহাইট / °F", "কেলভিন / K"}
        ComboboxInx3Combo2List = {"কিলোগ্রাম / kg", "সের Sher", "গ্রাম / g", "ভরি / Vori"}
        ComboboxInx4Combo2List = {"একর / Acre", "শতাংশ / Decimal", "বর্গফুট / Sq. ft"}
        MileMileKilo() 'local sub call only once during this load sub: initial display list in 3 comboBox
        ComboBox1.SelectedIndex = 0 'select 'length' catagory as default
        ComboBox2.SelectedIndex = 0 'select 'Mile' as Default
        ComboBox3.SelectedIndex = 1 'select 'kilometre' as dafault
        '==========================================
▯▯▯▯Show default conversion value in textBox1▯
        TextBox1.Text = "1"
        'Show default conversion result value in textBox2 -which is read only box
        TextBox2.Text = "1.6093"
        'Show default conversion realation in textBox3 -which is read only box
        TextBox3.Text = "১ মাইল = ১.৬০৯৩ কিলোমিটার"
        'Show default first value unit name in textBox4 & TextBox5, which are read only boxes
        TextBox4.Text = "মাইল" : TextBox5.Text = "কিলোমিটার"
        ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected iidex into global variable
        ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected iidex into global variable
        EnteredNumber = Val(TextBox1.Text) 'get the entred number into global variable
    End Sub
    'Any change will trigger calculation and display
    Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ComboBox1.SelectedIndexChanged
        CatItem = ComboBox1.SelectedIndex 'get the currently selected Index number into global variable
        'get selected items of ComboBox and Entered Number ino globals
        ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected index into global variable
        ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected index into global variable
        EnteredNumber = Val(TextBox1.Text)
        If CatItem = 0 Then Combo1One() 'sub call
        If CatItem = 1 Then AreaMileKilo() 'sub call
        If CatItem = 2 Then TempDegFahren() 'sub call
        If CatItem = 3 Then WeightKiloSher() 'sub call
        If CatItem = 4 Then SurveyFunction() 'sub call
    End Sub
    Private Sub ComboBox2_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ComboBox2.SelectedIndexChanged
        'get selected items of ComboBox and Entered Number ino globals
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected iidex into global variable
        ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected iidex into global variable
        EnteredNumber = Val(TextBox1.Text) : FromItem = ComboBox2.SelectedIndex
        TextBox3.Text = UpdateRelationDisplay(CatItem, FromItem, ToItem) 'function retuns String
        TextBox4.Text = GetLeftWord(TextBox3.Text) 'function call to display unit in textBox3
        TextBox5.Text = GetRightWord(TextBox3.Text) 'function call to display TextBox5
        'Find what has been changed and take action accordingly
        If ComboBox1.SelectedItem = "দূরত্ব / Length" Then
            CalculateAndDisplayLength() 'call local sub
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" Then
            CalculateAndDisplayArea() 'local sub call
        End If
        If ComboBox1.SelectedItem = "তাপ / Temperature" Then
            CalculateAndDisplayTemp() 'local sub call
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" Then
            CalculateAndDisplayWeight() 'local sub call
        End If
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" Then
            CalculateAndDisplaySurvey() 'local sub call
        End If
    End Sub
    Private Sub ComboBox3_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ComboBox3.SelectedIndexChanged
        'get selected item into global
        ToItem = ComboBox3.SelectedIndex
        'get selected items of ComboBox and Entered Number ino globals
        ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected iidex into global variable
        ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected iidex into global variable
        EnteredNumber = Val(TextBox1.Text)
        TextBox3.Text = UpdateRelationDisplay(CatItem, FromItem, ToItem) 'function retuns String
        TextBox4.Text = GetLeftWord(TextBox3.Text) 'function call to display unit in textBox3
        TextBox5.Text = GetRightWord(TextBox3.Text) 'runction call to display TextBox5
        'Find what has been changed and take action accordingly
        If ComboBox1.SelectedItem = "দূরত্ব / Length" Then
            CalculateAndDisplayLength() 'call local sub
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" Then
            CalculateAndDisplayArea() 'call local sub
        End If
        If ComboBox1.SelectedItem = "তাপ / Temperature" Then
            CalculateAndDisplayTemp() 'call local sub
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" Then
            CalculateAndDisplayWeight() 'call local sub
        End If
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" Then
            CalculateAndDisplaySurvey() 'call local sub
        End If
    End Sub
    'This Sub makes sure only number [0-9 and decimal point once] entered in TextBox1
    Private Sub TextBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress
        e.Handled = Not (Char.IsDigit(e.KeyChar) Or Asc(e.KeyChar) = 8 Or ((e.KeyChar = ".") And (sender.Text.IndexOf(".") = -1)))
    End Sub
'This Sub makes calculation as soon as the value entered into text box1
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Private Sub TextBox1_TextChanged(sender As Object, e As EventArgs) Handles TextBox1.TextChanged
    'get selected items of ComboBox and Entered Number ino globals
    ComBox2SelItemIdex = ComboBox2.SelectedIndex 'get the selected iidex into global variable
    ComBox3SelItemIndex = ComboBox3.SelectedIndex 'get the selected iidex into global variable
    EnteredNumber = Val(TextBox1.Text)
    If ComboBox1.SelectedItem = "দূরত্ব / Length" Then
        CalculateAndDisplayLength() 'call the 1st sub below
    End If
    If ComboBox1.SelectedItem = "ক্ষেত্র / Area" Then
        CalculateAndDisplayArea() 'call the 2nd sub below
    End If
    If ComboBox1.SelectedItem = "তাপ / Temperature" Then
        CalculateAndDisplayTemp() 'call the 3rd sub below
    End If
    If ComboBox1.SelectedItem = "ওজন / Weight" Then
        CalculateAndDisplayWeight() 'call the 4th sub below
    End If
    If ComboBox1.SelectedItem = "জমির হিসাব / Survey" Then
        CalculateAndDisplaySurvey() 'call the 5th Sub below
    End If
End Sub
'this sub adds catagory items in comboBox1
Sub MileMileKilo()
    Dim LoopContol As Integer
    ComboBox1.Items.Clear() 'must clear before adding list
    For LoopContol = 0 To 4
        ComboBox1.Items.Add(Combo1List(LoopContol))
    Next
End Sub
'this sub updates comboBox2 and ComboBox3 list Items, both box will have same list
'user chose length
Sub Combo1One()
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 6
        ComboBox2.Items.Add(ComboInx0Combo2List(LoopContol)) 'use declared array variable ComboInx1Combo2List
        ComboBox3.Items.Add(ComboInx0Combo2List(LoopContol)) 'use declared array variable ComboInx1Combo2List
    Next
    TextBox3.Text = "১ মাইল = ১.৬০৯৩ কিলোমিটার" show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list
End Sub
'user chose Area- this sub populates combo1 and 2
Sub AreaMileKilo()
    Dim LoopContol As Integer
    'Add Categories in the ComboBox2 and ComboBox3, both box will have same list
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 6
        ComboBox2.Items.Add(ComboInx1Combo2List(LoopContol)) 'use declared array variable ComboInx1Combo2List
        ComboBox3.Items.Add(ComboInx1Combo2List(LoopContol)) 'use declared array variable ComboInx1Combo2List
    Next
    TextBox3.Text = "১ বর্গমাইল = ১ বর্গমাইল" show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list
End Sub
'user chose Temperature- this sub populates combo1 and 2
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Sub TempDegFahren()
    Dim LoopContol As Integer
    'Add Categories in the ComboBox2 and ComboBox3, both will have same list
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 2
        ComboBox2.Items.Add(ComboboxInx2Combo2List(LoopContol)) 'use declared array variable ComboInx2Combo2List
        ComboBox3.Items.Add(ComboboxInx2Combo2List(LoopContol)) 'use declared array variable ComboInx2Combo2List
    Next
    TextBox3.Text = "১ ডিগ্রী.সেলসিয়াস = ১ ডিগ্রী.সেলসিয়াস" 'show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list
End Sub
'user chose Weight- this sub populates combo1 and 2
Sub WeightKiloSher()
    Dim LoopContol As Integer
    'Add Categories in the ComboBox2 and ComboBox3, both will have same list
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 3
        ComboBox2.Items.Add(ComboboxInx3Combo2List(LoopContol)) 'use declared array variable ComboInx3Combo2List
        ComboBox3.Items.Add(ComboboxInx3Combo2List(LoopContol)) 'use declared array variable ComboInx3Combo2List
    Next
    TextBox3.Text = "১ কিলোগ্রাম = ১ কিলোগ্রাম" 'show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list

End Sub
'user chose Survey- this sub populates combo1 and 2
Sub SurveyFunction()
    Dim LoopContol As Integer
    'Add Categories in the ComboBox2 and ComboBox3, both will have same list
    ComboBox2.Items.Clear() 'clear the list first
    ComboBox3.Items.Clear() 'clear the list first
    For LoopContol = 0 To 2
        ComboBox2.Items.Add(ComboboxInx4Combo2List(LoopContol)) 'use declared array variable ComboInx4Combo2List
        ComboBox3.Items.Add(ComboboxInx4Combo2List(LoopContol)) 'use declared array variable ComboInx4Combo2List
    Next
    TextBox3.Text = "১ একর = ১ একর" 'show relation
    ComboBox2.SelectedIndex = 0 'select default item of the list
    ComboBox3.SelectedIndex = 0 'select default item of the list
End Sub
'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
Sub CalculateAndDisplayLength()
    'Length category have 7 items
    If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then 'make sure there is a number in the entry text box
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "মাইল / Mile (mi)" Then
            TextBox2.Text = CalculatedValueMileTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module
CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "কিলোমিটার / Kilometre (km)" Then
            TextBox2.Text = CalculatedValueKiloMetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module
CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "মিটার / Metre (m)" Then
            TextBox2.Text = CalculatedValueMetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module
CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "মিলিমিটার / Millimetre (mm)" Then
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
            TextBox2.Text = CalculatedValueMilliMetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "গজ / Yard (yd)" Then
            TextBox2.Text = CalculatedValueYardTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "ফুট / Foot (ft)" Then
            TextBox2.Text = CalculatedValueFtTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module CalculationModule
        End If
        If ComboBox1.SelectedItem = "দূরত্ব / Length" And ComboBox2.SelectedItem = "ইঞ্চি / Inch (in)" Then
            TextBox2.Text = CalculatedValueInchTo(EnteredNumber, ComBox3SelItemIndex) 'function call which is in module CalculationModule
        End If
    End If
End Sub
'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
Sub CalculateAndDisplayArea()
    If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গমাইল / Sq. mi" Then
            TextBox2.Text = CalculatedValueSqMileTo(EnteredNumber, ComBox3SelItemIndex) 'function call. Result displayed in textBox2
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গকিলোমিটার / Sq. Km" Then
            TextBox2.Text = CalculatedValueSqKilometreTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গমিটার / Sq. mt" Then
            TextBox2.Text = CalculatedValueSqMetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গমিলিমিটার / Sq. mm" Then
            TextBox2.Text = CalculatedValueSqMillimetreTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গগজ / Sq. yd" Then
            TextBox2.Text = CalculatedValueSqYardTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গফুট / Sq. ft" Then
            TextBox2.Text = CalculatedValueSqFtTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ক্ষেত্র / Area" And ComboBox2.SelectedItem = "বর্গইঞ্চি / Sq. in" Then
            TextBox2.Text = CalculatedValueSqinchTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
    End If
End Sub
'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
Sub CalculateAndDisplayTemp()
    If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then
        If ComboBox1.SelectedItem = "তাপ / Temperature" And ComboBox2.SelectedItem = "ডিগ্রী.সেলসিয়াস / °C" Then
            TextBox2.Text = CalculatedValueCelciusTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "তাপ / Temperature" And ComboBox2.SelectedItem = "ডিগ্রী.ফারেনহাইট / °F" Then
            TextBox2.Text = CalculatedValueFahrenheitTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "তাপ / Temperature" And ComboBox2.SelectedItem = "কেলভিন / K" Then
            TextBox2.Text = CalculatedValueKelvinTo(EnteredNumber, ComBox3SelItemIndex) 'function call
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        End If
     End If
  End Sub
  'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
  Sub CalculateAndDisplayWeight()
     If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then
        If ComboBox1.SelectedItem = "ওজন / Weight" And ComboBox2.SelectedItem = "কিলোগ্রাম / kg" Then
           TextBox2.Text = CalculatedValueKgTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" And ComboBox2.SelectedItem = "সের Sher" Then '===
           TextBox2.Text = CalculatedValueSherTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" And ComboBox2.SelectedItem = "গ্রাম / g" Then
           TextBox2.Text = CalculatedValueGramTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "ওজন / Weight" And ComboBox2.SelectedItem = "ভরি / Vori" Then
           TextBox2.Text = CalculatedValueVoriTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
     End If
  End Sub
  'this sub is called by TextBox1_TextChanged, comboBox2 and ComboBox3 selectedIndexChanged Event sub
  Sub CalculateAndDisplaySurvey()
     If TextBox1.Text <> "" And IsNumeric(TextBox1.Text) = True Then
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" And ComboBox2.SelectedItem = "একর / Acre" Then
           TextBox2.Text = CalculatedValueAcreTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" And ComboBox2.SelectedItem = "শতাংশ / Decimal" Then
           TextBox2.Text = CalculatedValueDecimalTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
        If ComboBox1.SelectedItem = "জমির হিসাব / Survey" And ComboBox2.SelectedItem = "বর্গফুট / Sq. ft" Then
           TextBox2.Text = CalculatedValueSurveySqFtTo(EnteredNumber, ComBox3SelItemIndex) 'function call
        End If
     End If
  End Sub
End Class
```

শেষ হলো ফর্মের ক্লাসের কোড। এবার আমরা লাইন থেকে লাইনের পূর্ণাঙ্গ ব্যাখ্যা তুলে ধরছি। শিক্ষার্থীরা যে অংশ বুঝে নিয়েছেন তা পাঠ না করে পরবর্তীতে যেতে পারেন। তবে না বুঝলে প্রয়োজনে বার বার পাঠ করে বুঝার চেষ্টা করবেন।

১. প্রথমেই আমরা বেশ ক'টি গ্লোবাল [সর্বত্র প্রাপ্য] ভেরিয়েবল ডিক্লায়ার করেছি। ক্লাসের প্রথমেই এগুলো ডিক্লায়ার করতে হয়। এদের মধ্যে আছে: (১) ৫টি ইন্টেজার টাইপ: (ক) CatItem -ক্যাটাগরি আইটেমের ইনডেক্স, (খ) FromItem -'হতে' আইটেমের ইনডেক্স, (গ) ToItem -'টু' আইটেমের ইনডেক্স, (ঘ) `ComBox2SelItemIdex` -ব্যবহারকারী কর্তৃক সিলেক্ট করা দ্বিতীয় কম্বো বাক্সের ইনডেক্স সংখ্যা এবং `ComBox3SelItemIndex` -ব্যবহারকারী কর্তৃক সিলেক্ট করা তৃতীয় কম্বো বাক্সের ইনডেক্স সংখ্যা। (২) ১টি ডবল টাইপ: `EnteredNumber` -ব্যবহারকারী কর্তৃক প্রবেশকৃত সংখ্যা। (৩) ৬টি স্ট্রিং টাইপ এ্যারে: (ক) Combo1List(4) -কম্বো বাক্স ১-এর ০-৪ এই ৫টি ক্যাটাগরির আইটেমের লিস্ট, (খ) ComboInx0Combo2List(6) - কম্বো বাক্স ২-এর ০-৬ এই ৭টি আইটেমের (০ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট, (গ) ComboInx1Combo2List(6) - কম্বো বাক্স ২-এর ০-৬ এই ৭টি আইটেমের (১ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট, (ঘ) ComboInx2Combo2List(2) - কম্বো বাক্স ২-এর ০-২ এই ৩টি আইটেমের (২ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট, (ঙ) ComboInx3Combo2List(3) - কম্বো বাক্স ২-এর ০-৩ এই ৪টি আইটেমের (৩ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট এবং

১৪তম পরিচ্ছেদ: ***Complete Application Tutorials***

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

(চ) ComboInx4Combo2List(2) - কম্বো বাক্স ২-এর ০-২ এই ৩টি আইটেমের (৪ ক্যাটাগরির আইটেম মুতাবিক) লিস্ট।

লক্ষ করুন, উক্ত ৬টি স্ট্রিং এ্যারের কাজ হলো ফর্ম লোড হওয়ার সময় ডিফোল্ট লিস্ট হিসাবে তিনটি কম্বো বাক্সে প্রদর্শন করা। সুতরাং আপনি প্রথমেই দেখতে পাবেন প্রথম কম্বো বাক্সে আছে চারটি ক্যাটাগরির লিস্ট- এর মধ্যে প্রথমটি হবে দূরত্ব। একই সময় কম্বো লিস্ট ২-এ থাকবে 'মাইল' ও কম্বো লিস্ট ৩-এ থাকবে 'কিলোমিটার'। তবে আপনি যে কোনো ক্যাটাগরি প্রথম কম্বোতে বদলালে অন্যগুলোও পরিবর্তন হবে সঠিকভাবে। ডিফোল্ট লিস্ট আসবে ফর্ম লোড ইভেন্ট থেকে আর পরিবর্তন হবে কম্বো বাক্সের সিলেক্টেডইনডেক্সচেঞ্জড্ [SelectedIndexChanged] সিস্টেম ইভেন্ট সাব থেকে। উক্ত সব ভেরিয়েবল বিভিন্ন সাবে ব্যবহৃত হবে- এজন্যই গ্লোবাল [সর্বত্র প্রাপ্য] হিসাবে ডিক্লায়ার করা হয়েছে। এবার প্রথম সাবের ব্যাখ্যা আসবে।

১. সিস্টেম ইভেন্ট সাব: Form1_Load -এর ব্যাখ্যা: এ সাবের উদ্দেশ্য হলো সর্বপ্রথম ফর্মে যা কিছু প্রদর্শিত হবে তার নির্দেশনা। প্রথমত, আমরা উপরে গ্লোবাল হিসেবে ডিক্লায়ার করা ৬টি এ্যারেতে ৬টি লিস্ট প্রবেশ করেছি। এরপর আমরা তিনটি কম্বো বাক্সের প্রথমটিতে ক্যাটাগরি, দ্বিতীয় ও তৃতীয়টিতে ডিফোল্ট একক লিস্ট প্রদর্শনের নির্দেশ দিয়েছে। এটা করতে যেয়ে ব্যবহার করেছি ফর্মের ক্লাস ইভেন্ট সাবের অন্যত্র নিজেদের সৃষ্ট একটি সাব ব্যবহার করে, যার নাম MileMileKilo()। সুতরাং এ পর্যন্ত কোড রান করার পর তিনটি কম্বো বাক্সে তিনটি লিস্ট লিখা হবে। প্রথম বাক্সে সর্বদা একই লিস্ট থাকবে- কারণ এটা হলো ক্যাটাগরি লিস্ট। এর ডিফোল্ট [দৃশ্যমান লেখা] আইটেম হবে 'দূরত্ব / Length'। সুতরাং দ্বিতীয় ও তৃতীয়টিতে ডিফোল্ট হিসাবে থাকবে মাইল-কিলোমিটার। লক্ষ করুন, - কোন্ ইনডেক্টকৃত লেখা কম্বো বাক্সে থাকবে তা নির্ধারণ করতে নির্দেশ দিয়েছি এই কোডগুলো দ্বারা: `ComboBox1.SelectedIndex = 0`, `ComboBox2.SelectedIndex = 0` এবং `ComboBox3.SelectedIndex = 1`। অর্থাৎ ডিফোল্ট হিসেবে প্রদর্শিত হবে যথাক্রমে: দূরত্ব / Length, মাইল / Mile (mi) এবং কিলোমিটার / Kilometer (km)। টেক্সট্ বাক্স ১, ২, ৩, ৪ ও ৫-এ ডিফোল্ট মানের সাথে সামঞ্জস্য রেখে কিছু তথ্য ব্যবহারকারীকে দেওয়া হয়েছে। সাবের শেষের তিনটি লাইনের কাজ হলো কম্বো বাক্স ২ ও ৩ এর ইনডেক্স সংখ্যা এবং প্রবেশ ফিল্ডের সংখ্যাটি ডিক্লায়ার করা গ্লোবাল ভেরিয়েবলে ঢুকানো। উল্লেখ্য এই ঢুকানোর কাজ হবে যতোবারই ইনডেক্স ও সংখ্যা পরিবর্তন হবে। শেষ হলো Form1_Load -সাব এর ব্যাখ্যা।

২-৪. সিস্টেম ইভেন্ট সাব (২, ৩ ও ৪): -এর ব্যাখ্যা: এগুলো তিনটি কম্বো বাক্সের SelectedIndexChanged -সিস্টেম ইভেন্ট সাব। ব্যবহারকারী এ তিনটি বাক্সের আইটেম মুক্তভাবে চয়েজ করবেন। আমাদের কাজ হলো কোন্টি চয়েজ করেছেন তা জানা ও সে মুতাবিক ক্রিয়া সম্পাদন করা।
সাব ২-এর প্রথম ৩ লাইন দ্বারা CatItem, `ComBox2SelItemIdex`, এবং `ComBox3SelItemIndex` -এ তিনটি গ্লোবাল হিসেবে ডিক্লায়ার করা ভেরিয়েবলে মান ঢুকিয়েছি। ফলে এগুলো যে কোনো স্থান ও সাব থেকে কাজে লাগানো যাবে। সাবের CatItem ভেরিয়েবল ইনডেক্ট [0-8] বলে দেবে কোন্ ক্যাটাগরি চয়েজ করা হয়েছে। যে'টি চয়েজ হয়েছে সে মুতাবিক যথাক্রমে ৫টি সাবরুটিনের [Combo1One(), AreaMileKilo(), TempDegFahren(), WeightKiloSher(), SurveyFunction()] সঠিকটি কল করা হয়েছে। এ প্রতিটি সাবের কাজ হলো কম্বো বাক্সে সঠিক লিস্ট লেখা ও ডিফোল্ট একটি আইটেম দেখানো। আমরা এ সাবগুলোর ব্যাখ্যার সময় আরো বিস্তারিত বলবো। বাকী দু'টো ভেরিয়েবল ক্যালকুলেশনের জন্য ব্যবহৃত হয়েছে।
সাব ৩-এর প্রথম তিন লাইনে উপরে বর্ণিত ভেরিয়েবলগুলো পুনরায় আপডেট করা হয়েছে- কারণ নতুনভাবে এক বা একাধিক সংখ্যা ও মান পরিবর্তন হয়েছে। চতুর্থ লাইন দ্বারা `FromItem`-নামক গ্লোবাল ইন্টেজার ভেরিয়েবলে দ্বিতীয় কম্বো বাক্সের ইনডেক্স মান ঢুকানো হয়েছে। পরের তিনটি লাইনের কাজ হলো যথাক্রমে টেক্সট্ বাক্স ৩, ৪ ও ৫-এ সম্পর্ক, কোনটি থেকে একক পরিবর্তন হবে এবং কোন্ অপর এককে পরিবর্তন হওয়া উদ্দেশ্য তা প্রদর্শন করা।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

অবশ্য এ কাজের জন্য AllDisplayFunctionsModule.vb -এ ডিফাইন করা তিনটি ফাংশন তথা UpdateRelationDisplay, GetLeftWord এবং GetRightWord-কে কল করা হয়েছে। পরের চারটি -ব্লকে আমরা কম্বো বাক্স-১-এ কোন্ ক্যাটাগরি চয়েজ করা আছে তা পরীক্ষা করে যথাযথ তথ্য প্রদর্শনের জন্য নিচে ডিফাইন করা সাবরুটিন কল করেছি। এসব সাবরুটিন সম্পর্কে ব্যাখ্যা পরে আসবে।
সাব ৪-এর কাজ উপরের সাবের মতোই। সুতরাং এ সম্পর্কে অতিরিক্ত কিছু বলার নেই- শুধুমাত্র এটুকু যে, এটা হলো কম্বো বাক্স ৩-এর SelectedIndexChanged -ইভেন্ট সাব। সুতরাং সে মুতাবিক সবকিছু আবার আপডেটও করতে হবে।

৫. সিসেটম ইভেন্ট সাব: TextBox1_KeyPress -এর ব্যাখ্যা: টেক্সড্ বাক্স ১-এ ব্যবহারকারী সংখ্যা প্রবেশ করবেন। কিন্তু সংখ্যা ছাড়া অন্য কোনো অক্ষর বা ক্যারাক্টারও ঢুকাতে পারেন। ফলে অঙ্ক কষার নির্দেশনার সময় প্রোগ্রাম অভিযোগ তুলবে- সংখ্যা ছাড়া অন্য কিছু কেনো ঢুকানো হলো? এ সাব নিশ্চিত করে, যাতেকরে প্রোগ্রাম কোনো অভিযোগ তুলতে না পারে। শুধুমাত্র আস্ত ও দশমিক সংখ্যা ছাড়া অন্য কোনো কিছুই এ টেক্টড্ বাক্সে যাতে ঢুকানো যায় না, সেটা নিশ্চিত করতে যেয়ে এই কোড।

৬. সিস্টেম ইভেন্ট সাব: TextBox1_TextChanged -এর ব্যাখ্যা: টেক্সট্ বাক্সের এই ইভেন্ট ধরে আমরা ক্যালকুলেশন করে ফলাফল প্রদর্শন করেছি। অর্থাৎ সংখ্যা ঢুকানোর পরই এই সাব সক্রিয় হয়ে ওঠে। প্রথমেই সংগ্রহ করেছি তিনটি গুরুত্বপূর্ণ গ্লোবাল মান, যথা: ComBox2SelItemIdex, ComBox3SelItemIdex এবং EnteredNumber। এগুলো দ্বারাই ক্যালকুলেশন করবো। এরপর আমরা পাঁচটি If .... End IF টেস্ট দ্বারা সঠিকটি নির্ধারণ করে নিচের ৫টি [CalculateAndDisplayLength(), CalculateAndDisplayArea(), CalculateAndDisplayTemp(), CalculateAndDisplayWeight(), CalculateAndDisplaySurvey()] সাবরুটিনের একটি কল করেছি ক্যালকুলেশন করে ফলাফল প্রদর্শনের জন্য। লক্ষ করুন, ৫টি টেস্টের উদ্দেশ্য হলো কোন্ ক্যাটাগরি তা জানা ও সে মুতাবিক সাবরুটিন কল করা। ঐ ৫টি সাবরুটিনের ব্যাখ্যাও একে একে এখন তুলে ধরছি।

৭. আমাদের সৃষ্ট সাব: MileMileKilo() -এর ব্যাখ্যা: আগেই বলেছি ফর্মের লোড ইভেন্ট সাব [১ম সাব] থেকে এটা কল করা হয়েছে। এটার কাজ হলো কম্বো বাক্স-১ এর লিস্ট তৈরি করা। লক্ষ করুন, লোড সাবে ডিফাইন করা Combo1List নামক এ্যারে ভেরিয়েবলটি ব্যবহার করেছি লুপে। ইনডেক্স ০-৪ দ্বারা লিস্ট বানিয়েছি ঐ ভেরিয়েবলে সংরক্ষিত আইটেম নাম দ্বারা। সাবে মোট ৫টি মাত্র লাইন আছে। প্রথম লাইনে LoopContol নামক একটি ইন্টেজার ভেরিয়েবল ডিক্লায়ার করেছি। দ্বিতীয় লাইন দ্বারা কম্বো বাক্সের আগের আইটেম লিস্ট মুছে দেওয়া হয়েছে। এরপর ফর-নেক্সট্ লুপের মাধ্যমে পুরো লিস্টটি কম্বো বাক্সে যুক্ত করেছি।

৮. আমাদের সৃষ্ট সাব: Combo1One() -এর ব্যাখ্যা: এটাও উপরোক্ত (৭ম) সাবের মতো কাজ করে। অবশ্য এক্ষেত্রে পার্থক্য হলো কম্বো বাক্স ২ ও ৩-এর লিস্ট বানানো। এখানে বলা হচ্ছে, যেহেতু ব্যবহারকারী একটি ক্যাটাগরি চয়েজ করেছেন তাই সে মুতাবিক বাকী দু'টো বাক্সের লিস্টে কি হবে তাও আপডেট করা জরুরী। এ কাজটিই হলো এ সাবের উদ্দেশ্য। লক্ষ করুন, উভয় কম্বো বাক্সের লিস্ট ফর-নেস্ট লুপের মাধ্যমে বানানোর পর যে দু'টো সিলেক্ট করে দেখানো আমাদের উদ্দেশ্য তা শেষের লাইন দু'টোতে নির্দেশ দিয়েছি। একই সময় টেক্সট্ বাক্স-৩-এ সম্পর্ক দেখিয়েছি- যা এ ক্ষেত্রে হলো: ১ মাইল = ১.৬০৯৩ কিলোমিটার।

৯. আমাদের সৃষ্ট সাব: AreaMileKilo() -এর ব্যাখ্যা: এটার কাজও উপরোক্ত সাবের মতো। তবে এখানে বলা হচ্ছে ব্যবহারকারী 'ক্ষেত্র / Area' ক্যাটাগরি চয়েজ করেছেন, তাই সে মুতাবিক বাকী দুটো বাক্সের লিস্ট তৈরি করা দরকার।

১০. আমাদের সৃষ্ট সাব: TempDegFahren() -এর ব্যাখ্যা: এ সাবের কাজও অনুরূপ। এ ক্ষেত্রে যেহেতু ব্যবহারকারী 'তাপ / Temperature' চয়েজ করেছেন তাই সে মুতাবিক বকী দু'টো বাক্সের লিস্ট তৈরি করতে বলা হয়েছে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

১১. আমাদের সৃষ্ট সাব: WeightKiloSher() -এর ব্যাখ্যা: এটাও একইভাবে কাজ করে। এখানে যেহেতু চয়েজ করা হয়েছে 'ওজন / Weight' তাই সে মুতাবিক বাকী দু'টো বাক্সের লিস্ট তৈরির নির্দেশ দেওয়া হয়েছে।

১২. আমাদের সৃষ্ট সাব: SurveyFunction() -এর ব্যাখ্যা: এটার কাজ হলো 'জমির হিসাব / Survey' মুতাবিক বাকী দু'টো বাক্সের লিস্ট বানানো।

১৩. আমাদের সৃষ্ট সাব: CalculateAndDisplayLength() -এর ব্যাখ্যা: এ সাবও পরবর্তী আরো ৪টি সাবের ক্রিয়া অনুরূপ। সুতরাং প্রথম এ সাব সম্পর্কে বিস্তারিত ব্যাখ্যা তুলে ধরাই যথেষ্ট মনে করছি। এ সাবটি ডাকা হয় দ্বিতীয় ও তৃতীয় কম্বো বাক্সের সিলেকশন পরিবর্তন [selectedIndexChanged] ইভেন্ট সাব থেকে। এছাড়া এটি উপরোক্ত [৬ নং] সিস্টেম সাব থেকেও কল করা হয়। আমাদের উদ্দেশ্য হলো ব্যবহারকারী যে হিসাব চান তা যেনো তাড়াতাড়ি পেয়ে যান। সুতরাং আমরা লিস্ট পরিবর্তন ও সংখ্যা পরিবর্তন- এ উভয় সিস্টেম ইভেন্টকে ধরে ক্যালকুলেশন সেরে ফলাফল প্রদর্শনের ব্যবস্থা করেছি।
লক্ষ করুন, প্রতিটি সাব শুরু হয়েছে একটি If .... End If টেস্ট দ্বারা। এর ভেতর আবার আরো কিছু If .... End If টেস্ট আছে। বাইরের টেস্ট দ্বারা নিশ্চিত করা হচ্ছে যে, সংখ্যা টেক্স বাক্স খালি নয় এবং এর মধ্যে প্রবেশকৃত সংখ্যাটি গ্রহণযোগ্য। এরপর ভেতরের টেস্ট দ্বারা নির্ধারিত হয়েছে ক্যাটাগরি 'দূরত্ব / Length' এবং দ্বিতীয় কম্বো বাক্সে "মাইল / Mile (mi)" ইউনিট সিলেক্ট করা আছে কি না। যদি উভয় কন্ডিশন সত্য হয় তাহলে আমরা কল করেছি 'CalculationModule.vb'-এ ডিফাইন করা 'CalculatedValueMileTo()' -নামক ফাংশনটি। এ ফাংশন সম্পর্কে পরে বলবো। এখানে এটুকু বলে রাখছি, এটির কাজ হলো দূরত্ব ক্যাগরির দ্বিতীয় কম্বোতে 'মাইল' এর মাত্রা [সংখ্যা] কোন্ দূরত্ব ইউনিটে পরিবর্তন হবে তা নির্ধারণ করে ক্যালকুলেন করা ও ফলাফল ফেরত পাঠানো। লক্ষ করুন: TextBox2.Text = CalculatedValueMileTo() -এ লাইনটি একই সাথে দু'টি কাজ করছে: (ক) ফাংশন কল করছে ও ফেরত প্রাপ্ত সংখ্যাটি ডিসপ্লে ফিল্ডে [অর্থাৎ টেক্সড্ বাক্স-২ -এ] প্রদর্শনও করে দিচ্ছে। ঠিক অনুরূপ সাবরুটিনের বাকী ৬টি টেস্টও মাইল থেকে কোন্ ইউনিটে পরিবর্তিত হবে সেটা নিশ্চিত করে একই ফাংশন কল করে রেজাল্ট প্রদর্শনের কাজ করছে। মনে রাখবেন ফাংশন কল থেকে একটি মাত্র ভ্যালু রিটার্ন হবে- এবং তা নির্ধারিত হবে মাইল থেকে কোন্ ইউনিটে পরিবর্তন হবে সে মুতাবিক। দূরত্বের সাথে মোট ৭টি ইউনিট জড়িত থাকায় ৭টি আলাদা ফাংশন তৈরি করতে হয়েছে। অনুরূপ অন্যান্য ক্ষেত্রেও ক'টি ইউনিটের সঙ্গে প্রত্যেকটি ক্যাটাগরি জড়িত সে হিসাবে ফাংশনও বানিয়েছি।

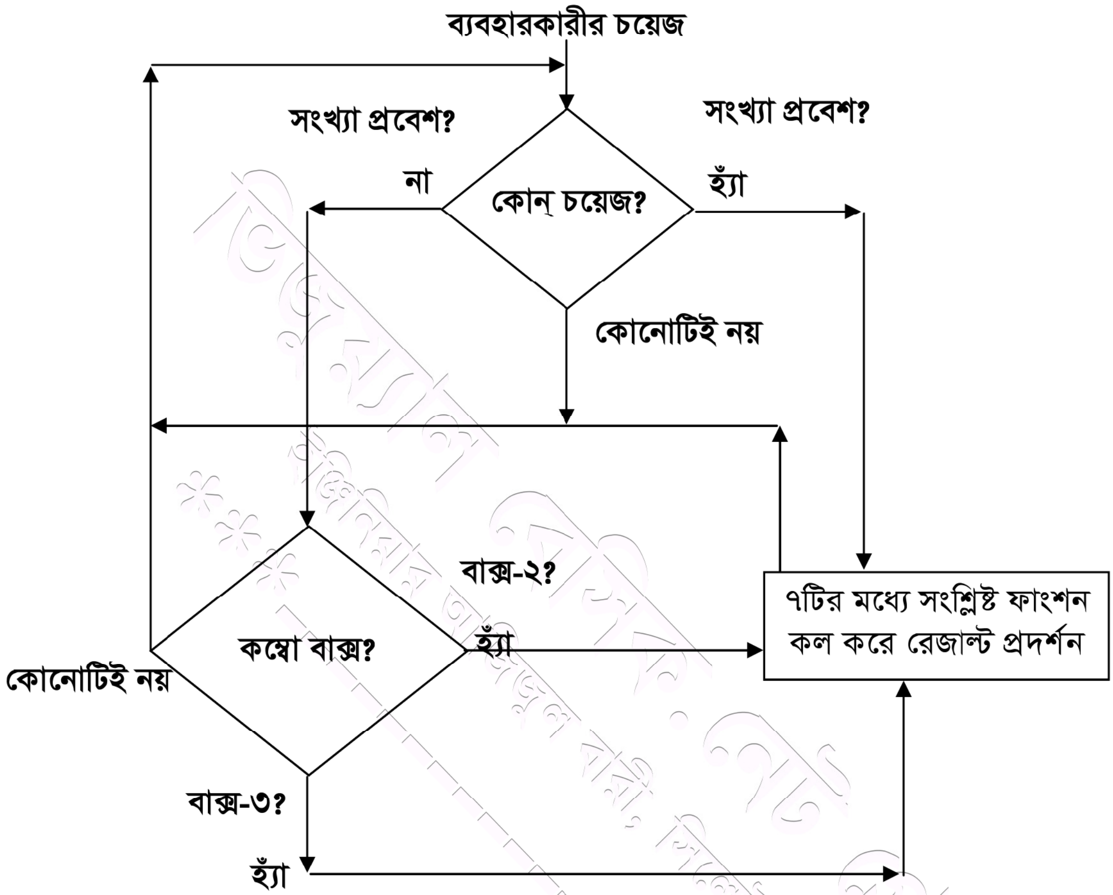
উপরের ব্যাখ্যা থেকে আমরা বুঝতে পারলাম যে, আমাদের সৃষ্ট ৫টি সাব [১৩ থেকে ১৭] মূলত টেক্সট বাক্সে সংখ্যা প্রবেশ কিংবা উভয় কম্বো বাক্সে [১ ও ২ -এর যে কোনোটিতে] আইটেম সিলেক্ট করার পরই কাজ শুরু করে। কোনটি সক্রিয়া হবে তা নির্ভর করে কোন্ আইটেম ব্যবহারকারী সিলেক্ট করছেন বা [সংখ্যা প্রবেশ করার ক্ষেত্রে] কোনটি সিলেক্ট করেছিলেন।

শিক্ষার্থীদের সুবিধার্থে পরের পৃষ্ঠায় উপরোক্ত ৫টি ফাংশনের ক্রিয়ার একটি ফ্লো চার্ট [পরম্পরা ক্রিয়াচিত্র] উপস্থাপন করেছি। এটা ভালো করে স্টাডি করে নিলে বিষয়টি সহজেই বুঝে এসে যাবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

বর্ণিত ৭টি সাবরুটিনের পরম্পরা ক্রিয়াচিত্র [ফ্লো চার্ট]



উপরের পরম্পরা ক্রিয়াচিত্র আমাদের প্রোগ্রামের এ্যালগোরিথম ও লজিক স্পষ্ট করেছে। ফর্ম প্রদর্শের পর আমরা অপেক্ষায় থাকি ব্যবহারকারীর পক্ষ থেকে ইনপুটের। তিনি হয় কোনো কম্বো বাক্সের লিস্ট বদলাবেন তার চাহিদা মতো না হয় টেক্সট ফিল্ড-১ -এ কোনো সংখ্যা লিখবেন। উভয় ক্রিয়া সম্পাদনকে আমরা কাজে লাগিয়েছি। আপনি যদি উপর থেকে এ্যারোগুলো অনুসরণ করেন তাহলে লজিকটি বুঝতে পারবেন। তিনটি ক্রিয়ার যে কোনো একটি ব্যবহারকারীকে করতেই হবে: যথা- ১. কোনো ক্রিয়া করবেন না। ২. প্রথমে কম্বো বাক্সের লিস্ট বদলিয়ে নিজের কাঙ্ক্ষিত একক থেকে একক পরিবর্তনের জন্য আইটেম চয়েজ করবেন এবং ৩. শেষে অথবা প্রথমেই ফিল্ডে সংখ্যাটি লিখবেন। প্রোগ্রাম বন্ধ করা ছাড়া তার পক্ষ থেকে আর কোনো ক্রিয়া সম্পাদনের পথ খোলা নেই।

এবার আমরা একটি কল্পিত অবস্থা নিয়ে ভাবতে পারি। যেমন: ১. প্রোগ্রাম চালু হাওয়ার পর ব্যবহারকারী ‘ওজন / Weight’ ক্যাটাগরি চয়েজ করলেন। এরপর দ্বিতীয় কম্বোতে যেয়ে চয়েজ করলেন ‘ভরি’ ও তৃতীয়টিতে যেয়ে চয়েজ করলেন ‘গ্রাম’। বুঝাই যাচ্ছে তিনি ভরি থেকে গ্রামের হিসাবটা জানতে চান। সবশেষে তিনি প্রবেশ করলেন কতো ভরি-তথা একটি সংখ্যা। উপরের ফ্লো চার্টে আমাদের প্রথম প্রশ্নের জবাব হবে ‘না’- এর পরের প্রশ্নের জবাব আসবে ‘হ্যাঁ’। এরপর চতুর্ভুজের ভেতরের ক্রিয়াগুলো সম্পাদিত হবে। যাক, আশাকরি পাঠকরা পরম্পরা ক্রিয়াচিত্র থেকে কিছুটা উপকৃত হয়েছেন। তবে প্রোগ্রামের অনেক কোড সম্পর্কে এখনো আমাদের জানার আছে। তাহলে আসুন, ওগুলোর ওপর বিস্তারিত বিবরণে যাই।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

২. প্রদর্শনের জন্য ফাংশন মডিউল [AllDisplayFunctionsModule.vb] -এর কোড

```
Public Module AllDisplayFunctionsModule
    'Declaration of arrays 'block1 - length to length
    Dim CatLenMileTo() As String = {"১ মাইল = ১ মাইল", "১ মাইল = ১.৬০৯৩ কিলোমিটার", "১ মাইল = ১৬০৯.৩৪৪ মিটার", "১ মাইল = ১৬০৯৩৪৪ মিলিমিটার", _
    "১ মাইল = ১৭৬০ গজ", "১ মাইল = ৫২৮০ ফুট", "১ মাইল = ৬৩৩৬০ ইঞ্চি"}
    Dim CatLenKiloTo() As String = {"১ কিলোমিটার = ১/১.৬০৯৩ মাইল", "১ কিলোমিটার = ১ কিলোমিটার", "১ কিলোমিটার = ১০০০ মিটার", "১ কিলোমিটার = ১০০০০০০ মিলিমিটার", _
   "১ কিলোমিটার = ১০৯৩.৬১৩৩ গজ", "১ কিলোমিটার = ৩২৮০.৮৩৯৯ ফুট", "১ কিলোমিটার = ৩৯৩৭০.০৭৮৮ ইঞ্চি"}
    Dim CatLenMeterTo() As String = {"১ মিটার = ১/১৬০৯.৩ মাইল", "১ মিটার = ১/১০০০ কিলোমিটার", "১ মিটার = ১ মিটার", "১ মিটার = ১০০০ মিলিমিটার", _
    "১ মিটার = ১.০৯৩৬ গজ", "১ মিটার = ৩.২৮০৮৪ ফুট", "১ মিটার = ৩৯.৩৭০০৭৮৮ ইঞ্চি"}
    Dim CatLenMilliMeterTo() As String = {"১ মিলিমিটার = ১/১৬০৯৩৪৪ মাইল", "১ মিলিমিটার = ১/১০০০০০০ কিলোমিটার", "১ মিলিমিটার = ১/১০০০ মিটার", "১ মিলিমিটার = ১ মিলিমিটার", _
    "১ মিলিমিটার  = ১/৯১৪.৪ গজ", "১ মিলিমিটার = ১/৩০৪.৮ ফুট", "১ মিলিমিটার = ১/২৫.৪ ইঞ্চি"}
    Dim CatLenYardTo() As String = {"১ গজ = ১/১৭৬০ মাইল", "১ গজ = ১/১০৯৩.৬১৩৩ কিলোমিটার", "১ গজ = ১/১.০৯৩৬ মিটার", "১ গজ = ৯১৪.৪ মিলিমিটার", _
    "১ গজ = ১ গজ", "১ গজ = ৩ ফুট", "১ গজ = ৩৬ ইঞ্চি"}
    Dim CatLenFootTo() As String = {"১ ফুট = ১/৫২৮০ মাইল", "১ ফুট = ১/৩২৮০.৮৩৯৯ কিলোমিটার", "১ ফুট = ১/৩.২৮০৮৪ মিটার", "১ ফুট = ৩০৪.৮ মিলিমিটার", _
    "১ ফুট = ১/৩ গজ", "১ ফুট = ১ ফুট", "১ ফুট = ১২ ইঞ্চি"}
    Dim CatLenInchTo() As String = {"১ ইঞ্চি  = ১/৬৩৩৬০ মাইল", "১ ইঞ্চি = ১/৩৯৩৭০.০৭৮৮ কিলোমিটার", "১ ইঞ্চি = ১/৩৯.৩৭০০৭৮৮৪ মিটার", "১ ইঞ্চি = ২৫.৪ মিলিমিটার", _
    "১ ইঞ্চি = ১/৩৬ গজ", "১ ইঞ্চি = ১/১২ ফুট", "১ ইঞ্চি = ১ ইঞ্চি"}
    '+++++++++++++ 'Block2 - Area to Area
    Dim CatAreaSqMileTo() As String = {"১ বর্গমাইল = ১ বর্গমাইল", "১ বর্গমাইল = ২.৫৯ বর্গকিলোমিটার", "১ বর্গমাইল = ১৮৬৮৩৪২৮.১১ বর্গমিটার", "১ বর্গমাইল = ২৫৮৯৯৮৮১১০৩৩৬ বর্গমিলিমিটার", _
   "১ বর্গমাইল = ৩০৯৭৬০০ বর্গগজ", "১ বর্গমাইল = ২৭৮৭৮৪০০ বর্গফুট", "১ বর্গমাইল = ৪০১৪৫০৫৬৫৮.০০৬৬ বর্গইঞ্চি"}
    Dim CatAreaSqKiloTo() As String = {"১ বর্গকিলোমিটার = ১/২.৫৯ বর্গমাইল", "১ বর্গকিলোমিটার = ১ বর্গকিলোমিটার", "১ বর্গকিলোমিটার = ১০০০০০০ বর্গমিটার", "১ বর্গকিলোমিটার = ১০০০০০০০০০০০০ বর্গমিলিমিটার", _
   "১ বর্গকিলোমিটার = ১১৯৫৯৯০.০৪৬৩ বর্গগজ", "১ বর্গকিলোমিটার = ১০৭৬৩৯১০.৪১৬৭ বর্গফুট", "১ বর্গকিলোমিটার = ১৫৫০০০৩১০০.০০৬২ বর্গইঞ্চি"}
    Dim CatAreaSqMeterTo() As String = {"১ বর্গমিটার = ১/১৮৬৮৩৪২৮.১১ বর্গমাইল", "১ বর্গমিটার = ১/১০০০০০০ বর্গকিলোমিটার", "১ বর্গমিটার = ১ বর্গমিটার", "১ বর্গমিটার = ১০০০০০০ বর্গমিলিমিটার", _
    "১ বর্গমিটার = ১.১৯৫৯৯ বর্গগজ", "১ বর্গমিটার = ১০.৭৬৩৯১০ বর্গফুট", "১ বর্গমিটার = ১৫৫০.০০৩১ বর্গইঞ্চি"}
    Dim CatAreaSqMilliMeterTo() As String = {"১ বর্গমিলিমিটার = ১/২৫৮৯৯৮৮১১০৩৩৬ বর্গমাইল", "১ বর্গমিলিমিটার = ১/১০০০০০০০০০০০০ বর্গকিলোমিটার", "১ বর্গমিলিমিটার = ১/১০০০০০০ বর্গমিটার", "১ বর্গমিলিমিটার = ১ বর্গমিলিমিটার", _
    "১ বর্গমিলিমিটার = ১/৮৩৬১২৭.৩৬ বর্গগজ", "১ বর্গমিলিমিটার = ১/৯২৯০৩.০৪ বর্গফুট", "১ বর্গমিলিমিটার =
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
১/৬৪৫.১৬ বর্গইঞ্চি"}
    Dim CatAreaSqYardTo() As String = {"১ বর্গগজ = ১/৩০৯৭৬০০ বর্গমাইল", "১ বর্গগজ = ১/১১৯৫৯৯০.০৮৬৩ বর্গকিলোমিটার", "১ বর্গগজ = ১/১.১৯৫৯৯ বর্গমিটার", "১ বর্গগজ = ৮৩৬১২৭.৩৬ বর্গমিলিমিটার", _
    "১ বর্গগজ = ১ বর্গগজ", "১ বর্গগজ = ৯ বর্গফুট", "১ বর্গগজ = ১২৯৬ বর্গইঞ্চি"}
    Dim CatAreaSqFootTo() As String = {"১ বর্গফুট = ১/২৭৮৭৮৪০০ বর্গমাইল", "১ বর্গফুট = ১/১০৭৬৩৯১০.৪১৬৭ বর্গকিলোমিটার", "১ বর্গফুট = ১/১০.৭৬৩৯১০৪১৬৭ বর্গমিটার", "১ বর্গফুট = ৯২৯০৩.০৮ বর্গমিলিমিটার", _
    "১ বর্গফুট = ১/৯ বর্গগজ", "১ বর্গফুট = ১ বর্গফুট", "১ বর্গফুট = ১৪৪ বর্গইঞ্চি"}
    Dim CatAreaSqInchTo() As String = {"১ বর্গইঞ্চি = ১/৪০১৪৪৫০৫৬৫৮.০০৬৬ বর্গমাইল", "১ বর্গইঞ্চি = ১/১৫৫০০০৩১০০.০০৬২ বর্গকিলোমিটার", "১ বর্গইঞ্চি = ১/১৫৫০.০০৩১ বর্গমিটার", "১ বর্গইঞ্চি = ৬৪৫.১৬ বর্গমিলিমিটার", _
    "১ বর্গইঞ্চি = ১/১২৯৬ বর্গগজ", "১ বর্গইঞ্চি = ১/১৪৪ বর্গফুট", "১ বর্গইঞ্চি = ১ বর্গইঞ্চি"}
    '+++++++++++++ Block3 - Temperature to Temperature
    Dim CatTempCelciusTo() As String = {"১ ডিগ্রী.সেলসিয়াস = ১ ডিগ্রী.সেলসিয়াস", "১ ডিগ্রী.সেলসিয়াস = (°C × 180/100) + 32 ) ডিগ্রী.ফারেনহাইট", "১ ডিগ্রী.সেলসিয়াস = (°C + 273.15) কেলভিন"}
    Dim CatTempFahrenheightTo() As String = {"১ ডিগ্রী.ফারেনহাইট = (°F - 32) × 100/180 ডিগ্রী.সেলসিয়াস", "১ ডিগ্রী.ফারেনহাইট = ১ ডিগ্রী.ফারেনহাইট", "১ ডিগ্রী.ফারেনহাইট = (°F - 32) × 5 / 9 + 273.15 কেলভিন"}
    Dim CatTempKelvinTo() As String = {"১ কেলভিন = (K - 273.15) ডিগ্রী.সেলসিয়াস", "১ কেলভিন = (K × 9/5) – 459.67 ডিগ্রী.ফারেনহাইট", "১ কেলভিন = ১ কেলভিন"}
'++++++++++++++ 'Block4 -
    Dim CatWeightKilogramTo() As String = {"১ কিলোগ্রাম = ১ কিলোগ্রাম", "১ কিলোগ্রাম = ১/০.৯৩৩ সের", "১ কিলোগ্রাম = ১০০০ গ্রাম", "১ কিলোগ্রাম = ১০০০/১১.৬৬ ভরি"}
    Dim CatWeightSherTo() As String = {"১ সের = ০.৯৩৩১০ কিলোগ্রাম", "১ সের = ১ সের", "১ সের = ৯৩৩.১ গ্রাম", "১ সের = ৮০.০১৭১৫২৬৫৯ ভরি"}
    Dim CatWeightGramTo() As String = {"১ গ্রাম = ১/১০০০ কিলোগ্রাম", "১ গ্রাম = ১/৯৩৩.১০ সের", "১ গ্রাম = ১ গ্রাম", "১ গ্রাম = ১/১১.৬৬ ভরি"}
    Dim CatWeightVoriTo() As String = {"১ ভরি = ১/৮৫.৭৬৩২৯৩৩ কিলোগ্রাম", "১ ভরি = ১/৮০.০১৭১৫২৬৫৯ সের", "১ ভরি = ১১.৬৬ গ্রাম", "১ ভরি = ১ ভরি"}
    '+++++++++++ 'Block5
    Dim CatSurveytAcreTo() As String = {"১ একর = ১ একর", "১ একর = ১০০ শতাংশ", "১ একর = ৪৩৫৬০.১৭৮২ বর্গফুট"}
    Dim CatSurveyDecimalTo() As String = {"১ শতাংশ = ১/১০০ একর", "১ শতাংশ = ১ শতাংশ", "১ শতাংশ = ৪৩৫.৬০১৭৪২ বর্গফুট"}
    Dim CatSurveySqFootTo() As String = {"১ বর্গফুট = ১/৪৩৫৬০.১৭৮২ একর", "১ বর্গফুট = ১/৪৩৫.৬০১৭৮২ শতাংশ", "১ বর্গফুট = ১ বর্গফুট"}
```

উপরে ফাংশনের শুরুর পূর্বে এ্যারে ডিক্লারেশন কোড দেখানো হয়েছে। সবগুলো এ্যারের মধ্যে নির্দিষ্ট কয়েকটি ইলেমেন্ট আছে [০-৬ দু'টি (এতে আছে ৭টি করে টেক্সট্ ইলেমেন্ট), ০-২ দু'টি (এতে আছে ৩টি করে টেক্সট্ ইলেমেন্ট), ০-৩ একটি (এতে আছে ৪টি টেক্সট্ ইলেমেন্ট)]। এই ইলেমেন্টগুলো আমরা বাংলা ইউনিকোডে লিখেছি। এগুলো ব্যবহারকারী কম্বো বাক্স পরিবর্তন করলেই সম্পর্ক টেক্সট্ বাক্সে দেখতে পাবেন। সুতরাং কোন্ একক থেকে কোন্ এককে বদলানো হচ্ছে ও তাদের মধ্যে সম্পর্ক কী তা সাথে সাথে প্রদর্শিত হবে। পরের পৃষ্ঠাসমূহে প্রদর্শনের ফাংশন কোডগুলো দেখানো হয়েছে। ব্যাখ্যা পরবর্তীতে এসেছে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
'this function retuns  the correct Relation text for displaying in
TextBox3 -Formula or Relation- uses 3 integer params
Public Function UpdateRelationDisplay(SuplIndx1 As Integer,
SuplIndx2 As Integer, SuplIndx3 As Integer) As String
    Dim LoopCont As Integer 'to cotrol loop
    '============Length Start==========
    If SuplIndx1 = 0 And SuplIndx2 = 0 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenMileTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 1 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenKiloTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 2 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenMeterTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 3 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenMilliMeterTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 4 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenYardTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 5 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenFootTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 0 And SuplIndx2 = 6 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenInchTo(LoopCont)
            End If
        Next
    End If
    '*****Length finished********
    '+++++++Area Start+++++++++
    If SuplIndx1 = 1 And SuplIndx2 = 0 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatAreaSqMileTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 1 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatAreaSqKiloTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 2 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatAreaSqMeterTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 3 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatAreaSqMilliMeterTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 4 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatAreaSqYardTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 5 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatAreaSqFootTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 1 And SuplIndx2 = 6 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatAreaSqInchTo(LoopCont)
            End If
        Next
    End If
    '*****Area finished****
    '+++Temperature Start++
    If SuplIndx1 = 2 And SuplIndx2 = 0 Then
        For LoopCont = 0 To 2
            If SuplIndx3 = LoopCont Then
                Return CatTempCelciusTo(LoopCont)
            End If
        Next
    End If
    If SuplIndx1 = 2 And SuplIndx2 = 1 Then
        For LoopCont = 0 To 2
            If SuplIndx3 = LoopCont Then
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
            Return CatTempFahrenheightTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 2 And SuplIndx2 = 2 Then
    For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
            Return CatTempKelvinTo(LoopCont)
        End If
    Next
End If
'==========Temperature Finished========
'============Weight Start=======
If SuplIndx1 = 3 And SuplIndx2 = 0 Then
    For LoopCont = 0 To 3
        If SuplIndx3 = LoopCont Then
            Rcturn CatWcightKilogramTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 3 And SuplIndx2 = 1 Then
    For LoopCont = 0 To 3
        If SuplIndx3 = LoopCont Then
            Return CatWeightSherTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 3 And SuplIndx2 = 2 Then
    For LoopCont = 0 To 3
        If SuplIndx3 = LoopCont Then
            Return CatWeightGramTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 3 And SuplIndx2 = 3 Then
    For LoopCont = 0 To 3
        If SuplIndx3 = LoopCont Then
            Return CatWeightVoriTo(LoopCont)
        End If
    Next
End If
'=========Weight Finished=======
'+++++++Survey Start+++++
If SuplIndx1 = 4 And SuplIndx2 = 0 Then
    For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
            Return CatSurveytAcreTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 4 And SuplIndx2 = 1 Then
    For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
            Return CatSurveyDecimalTo(LoopCont)
        End If
    Next
End If
If SuplIndx1 = 4 And SuplIndx2 = 2 Then
    For LoopCont = 0 To 2
        If SuplIndx3 = LoopCont Then
            Return CatSurveySqFootTo(LoopCont)
        End If
    Next
End If
'=========Survey Finished========
Return ""
End Function
'these two function gets the from unit word to display in
textBox4 ans TextBox5
Public Function GetLeftWord(StrVar As String) As String
    Dim AllWords As String() = StrVar.Split(New Char() {"
"})
    On Error GoTo LabelOne
    Return AllWords(1) 'this is the 2nd word which we need
LabelOne:
    Rcturn ""
End Function
'=========
Public Function GetRightWord(StrVar As String) As String
    Dim AllWords As String() = StrVar.Split(New Char() {"
"}) 'Split is a function which seperates words
    Dim A As Integer = (AllWords.Length) 'get total number
of words
    On Error GoTo LebelOne
    Return AllWords(A - 1) 'get the index of last word
LebelOne:
    Return ""
  End Function
End Module
=============================================
```

এখানেই শেষ হলো মডিউলে ডিফাইন-করা ফাংশনের কোড। লক্ষ করুন, পুরো মডিউলে মোট তিনটি মাত্র ফাংশন আছে। প্রথম বড়ো ফাংশনের কাজ হলো প্রদর্শন ফিল্ডে দু'টি এককের সম্পর্ক রিটার্ন করা। দ্বিতীয় ও তৃতীয়টির কাজ হলো ফিল্ড ৪ ও ৫-এ সঠিক একক নাম প্রদর্শন করা।

প্রথম ফাংশনের মধ্যে মোট ২৪টি 'ইফ-এন্ড ইফ' [If - End If] টেস্ট আছে। এগুলো একই ধরনের। সুতরাং একটি সম্পর্কে পূর্ণ ব্যাখ্যাই যথেষ্ট হবে।

ফাংশনের 'ইফ-এন্ড ইফ' টেস্ট কোড রান করার আগে 'ফর-নেক্সট' [For - Next] লুপ নিয়ন্ত্রণের জন্য প্রথমেই আমরা একটি ইন্টেজার ভেরিয়েবল ডিক্লায়ার করেছি। এই একই ভেরিয়েবল ২৪টি টেস্টেই ব্যবহৃত হয়েছে। এবার প্রথম 'ইফ-এন্ড ইফ' ব্লকটি আবার তুলে ধরছি।

```
Dim LoopCont As Integer 'to control loop
    '============Length Start==========
    If SuplIndx1 = 0 And SuplIndx2 = 0 Then
        For LoopCont = 0 To 6
            If SuplIndx3 = LoopCont Then
                Return CatLenMileTo(LoopCont)
            End If
        Next
    End If
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

প্রথম লাইনেই LoopCont নামক ইন্টেজার ভেরিয়েবল ডিক্লারেশনটি দেখা যাচ্ছে। দ্বিতীয় লাইনে শুরু হয়েছে টেস্ট: If SuplIndx1 = 0 And SuplIndx2 = 0 Then -এর অর্থ হলো যদি ইনডেক্স১ [প্রথম কম্বো বাক্সের ইনডেক্স] ০ এবং ইনডেক্স২ [দ্বিতীয় কম্বো বাক্সের ইনডেক্স] ০ হয় তাহলে পরের লাইনে যাও- অন্যথায় নয়।

উভয় কন্ডিশন সত্য হলে পরের লাইনে এসে 'ফর-নেস্ট' লুপটি শুরু হবে। এ লুপের কাজ হলো তৃতীয় কম্বো বাক্সের ইনডেক্স কি তা সনাক্ত করা। সুতরাং ০ থেকে ৬ পর্যন্ত- এই মোট ৭ বার লুপ রান করবে। প্রত্যেকবার দেখে নেবে কম্বো বাক্স-৩ এর ইনডেক্সের সাথে লুপ সংখ্যার মিল আছে কি না। মিল হলেই মাঝের লাইনে যেয়ে রিটার্ন ভ্যালু প্রেরণ করে ফাংশনের কোড রান হওয়ার ইতি ঘটবে। তবে কী রিটার্ন করবে? রিটার্ন করবে একটি স্ট্রিং যা আসবে CatLenMileTo(LoopCont) থেকে। মনে আছে কি, আমরা প্রথমেই ২৪টি এ্যারে ভেরিয়েবল ডিক্লায়ার করেছিলাম? CatLenMileTo-হলো এই এ্যারেগুলোর মধ্যে প্রথমটির নাম। তৃতীয় কম্বো বাক্সের ইনডেক্স সংখ্যা LoopCont-এর সমান। তাই এই ইনডেক্স মুতাবিক ঐ এ্যারের ইলেমেন্টই ফাংশন স্ট্রিং হিসেবে রিটার্ন করবে। কথাগুলো বুঝে না আসলে পুনরায় পাঠ করবেন।

প্রথম ব্লকের টেস্ট ফেইল করলে, দ্বিতীয় ব্লকে যেয়ে টেস্ট হবে- এভাবে যতক্ষণ কন্ডিশন মিলবে না ততক্ষণ টেস্ট করতে থাকবে। আর অবশ্যই, যে কোনো একটি টেস্টের ফলাফল সত্য হবে। কারণ, ব্যবহারকারী এই ২৪টি কম্বিনেশন ছাড়া বাইরে কিছু করতে পারবেন না। সুতরাং যেটাই টেস্ট থেকে সত্য ফলাফল দিক না কেনো, ফাংশন সঠিক টেক্সট রিটার্ন করবে। আর মনে রাখবেন, আমরা এই ফাংশনটি কল করেছি Form.vb এর ক্লাসের ভেতরস্থ ৩টি কম্বো বাক্স পরিবর্তন সিস্টেম ইভেন্ট সাব থেকে। অন্যকথায় ব্যবহাকারী এই তিনটি বাক্সের যে কোনটির ইনডেক্স পরিবর্তন করলেই ফাংশন কল করা হবে।

এবার শেষের দুটো ফাংশনের ওপর কিছুটা ব্যাখ্যার প্রয়োজন। এ দু'টো আবার দেখে নিই:

```
'Function uses StrVar as a string param- comes from calling sub
Public Function GetLeftWord(StrVar As String) As String
  'Split is a function which seperates words
  Dim AllWords As String() = StrVar.Split(New Char() {" "})
   On Error GoTo LabelOne
   Return AllWords(1) 'this is the 2nd word which we need
LabelOne:
    Return ""
 End Function
```

```
'Function uses StrVar as a string param- comes from calling sub
Public Function GetRightWord(StrVar As String) As String
   'Split is a function which seperates words
   Dim AllWords As String() = StrVar.Split(New Char() {" "})
   'get total number of words
  Dim A As Integer = (AllWords.Length)
  On Error GoTo LebelOne
  Return AllWords(A - 1) 'get the index of last word
LebelOne:
   Return ""
End Function
```

লক্ষ করুন, উভয় ফাংশনের একটি প্যারামিটার আছে। StrVar নামক এ প্যারামিটার আসবে যেখান থেকে এটি ডাক দেওয়া হয়েছে। অর্থাৎ আমরা সম্পর্ক বাক্সে প্রদর্শিত টেক্সট প্যারামিটার হিসাবে ব্যবহার করেছি। ফাংশন দু'টোর কাজ হলো বায়ের পূর্ণ শব্দ [যা হলো 'হতে' এককের নাম] এবং ডানের পূর্ণ শব্দ [যা হলো 'প্রতি' এককের নাম] বের করে রিটার্ন করা। প্রথম লাইনে 'AllWords' নামক একটি স্ট্রিং ভেরিয়েবল ডিক্লায়ার করেছি। সমান প্রতীকের ডানে প্যারামিটার স্ট্রিংকে 'Split(New Char() {" "})' নাম একটি সিস্টেম ফাংশন দ্বারা শব্দে শব্দে বিভক্ত করে নিয়েছি। এর পরের লাইনের কাজ হলো কোনো এরোর থাকলে অভিযোগ ছাড়াই ফাংশন থেকে বের হওয়ার ব্যবস্থা। অন্যথায় ফাংশন রিটার্ন করবে আমাদের কাঙ্ক্ষিত প্রথম শব্দটি [Return AllWords(1)]। দেখতেই পাচ্ছেন, উভয় ফাংশনের কাজ একই ধরনের। ডানেরটি রিটার্ন করবে শেষের শব্দটি [Return AllWords(A - 1)]। রিটার্ন হওয়া শব্দ দু'টো যথাক্রমে TextBox4 ও TextBox5-এ প্রদর্শিত হবে। সুতরাং আমাদের নিকট স্পষ্ট থাকবে কোন্ দু'টি একক নিয়ে কাজ হচ্ছে। শেষ হলো Public Module AllDisplayFunctionsModule -এর ব্যাখ্যা। এখন আমারেদকে বুঝতে হবে ক্যালকুলেশন মডিউলের কাজটি কিভাবে হয়।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

CalculationModule.vb-এ ডিফাইন করা ফাংশনগুলোর ওপর বিস্তারিত আলোচনা

ইতোমধ্যে যা কিছু কোড লিখিত হয়েছে তা সবই হলো ক্যালকুলেন সঠিকভাবে সম্পাদনের প্রস্তুতি মাত্র। অর্থাৎ কোন্ ফিল্ডে কি প্রদর্শন হবে, ব্যবহারকারী কোন্ ইউনিট থেকে কোন্ ইউনিটে পরিবর্তন করবেন, সংখ্যাটি কি, ইত্যাদি প্রশ্নের সঠিক জবাব আগের সকল কোড থেকে আমরা সুস্পষ্ট ও সঠিকভাবে সনাক্ত করেছি ও সংশ্লিষ্ট প্রদর্শন ফিল্ডে দেখানোরও ব্যবস্থা করেছি। এবার আসল কাজ তথা ক্যালকুলেশনের ক্রিয়াটি সম্পাদন করে রেজাল্ট দেখাতে হবে- আর এটাই তো পুরো প্রোগ্রামের আসল লক্ষ্য।

আমরা সিদ্ধান্ত করেছি, ক্যালকুলেন ও রেজাল্ট প্রদর্শন হবে একটি স্বয়ংক্রিয় কাজ। এর জন্য আলাদা কোনো বোতামে ক্লিক করার প্রয়োজন নেই। সুতরাং যে মুহূর্তে ব্যবহারকারী কোনো ক্রিয়া সম্পাদন করবেন- প্রোগ্রাম সাথে সাথে ফলাফল ও তথ্যাদি প্রদর্শন করবে- এটাই আমার চাই। আমরা জানি ব্যাবহারকারী মোট চারটি ক্রিয়ার যে কোনো একটি করতে পারেন। এগুলো হলো: ১. প্রথম কম্বো বাক্সের আইটেম পরিবর্তন [ক্যাটাগরি চয়েজ]। ২. দ্বিতীয় কম্বো বাক্সের আইটেম পরিবর্তন ['হতে' একক চয়েজ]। ৩. তৃতীয় কম্বো বাক্সের আইটেম পরিবর্তন ['প্রতি' একক চয়েজ]। ৪. সংখ্যা প্রবেশ ফিল্ড টেক্সট বাক্স-১ এ পরিবর্তন [সংখ্যা মান প্রবেশ]। আমাদের কোড এ চারটির মধ্যে কোন্‌টিতে রদবদল হয়েছে তা জেনে ক্যালকুলেশন করবে। এটা করতে যেয়ে আমরা পূর্বপ্রস্তুতি হিসেবে আগে ব্যাখ্যাত কোডগুলো কাজে লাগিয়েছি। এরপর যেখান থেকে যেভাবে প্রয়োজন CalculationModule.vb-এ ডিফাইন করা ফাংশনগুলোর সঠিকটি কল করে ফলাফল বের করে প্রদর্শন করেছি। কিভাবে? এক্ষুণি বুঝে নেবেন। তবে প্রথমে পুরো ক্যালকুলেন কোড ব্লকটি তুলে ধরা প্রয়োজন।

```
Public Module CalculationModule
    '========= Length Relation========
   Dim LengthRelation1() As Double = {1, 1.609344, 1609.344, 1609344, 1760, 5280, 63360}
    Dim LengthRelation2() As Double = {1 / 1.6093, 1, 1000, 1000000, 1093.6133, 3280.8399, 39370.0788}
    Dim LengthRelation3() As Double = {1 / 1609.3, 1 / 1000, 1, 1000, 1.0936133, 3.2808399, 39.3700788}
    Dim LengthRelation4() As Double = {1 / 1609344, 1 / 1000000, 1 / 1000, 1, 1 / 914.4, 1 / 304.8, 1 / 25.4}
    Dim LengthRelation5() As Double = {1 / 1760, 1 / 1093.6133, 1 / 1.0936, 914.4, 1, 3, 36}
    Dim LengthRelation6() As Double = {1 / 5280, 1 / 3280.8399, 1 / 3.28084, 304.8, 3, 1, 12}
    Dim LengthRelation7() As Double = {1 / 63360, 1 / 39370.0788, 1 / 39.37007884, 25.8, 1 / 36, 1 / 12, 1}
    '=========== Area Relation
    Dim AreaRelation8() As Double = {1, 2.59, 18683428.11, 2589988110336, 3097600, 27878400, 4014505658.0066}
    Dim AreaRelation9() As Double = {1 / 2.59, 1, 1000000, 1000000000000, 1195990.0463, 10763910.4167, 1550003100.0062}
    Dim AreaRelation10() As Double = {1 / 18683428.11, 1000000, 1, 1000000, 1.19599, 10.7639104167, 1550.0031}
    Dim AreaRelation11() As Double = {1 / 2589988110336, 1 / 1000000000000, 1 / 1000000, 1, 1 / 836127.36, 1 / 92903.04, 1 / 645.16}
    Dim AreaRelation12() As Double = {1 / 3097600, 1 / 1195990.0863, 1 / 1.19599, 836127.36, 1, 9, 1296}
    Dim AreaRelation13() As Double = {1 / 27878400, 1 / 10763910.4167, 1 / 10.7639104167, 92903.08, 1 / 9, 1, 144}
    Dim AreaRelation14() As Double = {1 / 4014505658.0066, 1 / 1550003100.0062, 1 / 1550.0031, 645.16, 1 / 1296, 1 / 144, 1}
    '=========== Weight Relation
    Dim WeightRelation15() As Double = {1, 1 / 0.933, 1000, (1000 / 11.66)}
    Dim WeightRelation16() As Double = {0.9331, 1, 933.1, 80.017152659}
    Dim WeightRelation17() As Double = {1 / 1000, 1 / 933.1, 1, 1 / 11.66}
    Dim WeightRelation18() As Double = {1 / (1000 / 11.6), 1 / 80.017152659, 11.66, 1}

    '=========== Survey Relation
    Dim SurveyRelation19() As Double = {1, 100, 43560.1782}
    Dim SurveyRelation20() As Double = {1 / 100, 1, 435.601782}
    Dim SurveyRelation21() As Double = {1 / 43560.1782, 1 / 435.601782, 1}
    'These functions returns Lengh Catagory to all combination calculated value. 2 parameters comes from calling sub
    Public Function CalculatedValueMileTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 6
            If ForCounter = ComBox3SelItemIndex Then
                Return LengthRelation1(ForCounter) * EnteredNumber
            End If
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueKiloMetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return LengthRelation2(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueMetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return LengthRelation3(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueMilliMetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return LengthRelation4(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueYardTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return LengthRelation5(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueFtTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return LengthRelation6(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueInchTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return LengthRelation7(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'These functions returns Area Catagory to all combination calculated value. 3 parameters comes from calling sub
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
Public Function CalculatedValueSqMileTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation8(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueSqKilometreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation9(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueSqMetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation10(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'This function returns Area, SqMillimetre to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueSqMillimetreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation11(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
Public Function CalculatedValueSqYardTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation12(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'This function returns Area, SqFt to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueSqFtTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
        If ForCounter = ComBox3SelItemIndex Then
            Return AreaRelation13(ForCounter) * EnteredNumber
        End If
    Next
    Return 0
End Function
'This function returns Area, SqInch to all combination calculated value. 3 parameters comes from calling sub
Public Function CalculatedValueSqinchTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
   Dim ForCounter As Integer
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        For ForCounter = 0 To 6
            If ForCounter = ComBox3SelItemIndex Then
                Return AreaRelation14(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
    'This function returns Temperature, celcius to all combination calculated value. 2 parameters comes from calling sub
    Public Function CalculatedValueCelciusTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        If ComBox3SelItemIndex = 0 Then Return EnteredNumber * 1
        If ComBox3SelItemIndex = 1 Then Return (EnteredNumber * 1.8) + 32
        If ComBox3SelItemIndex = 2 Then Return EnteredNumber + 273.15
        Return 0
    End Function
    'This function returns Temperature, Fahrenheit to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueFahrenheitTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        If ComBox3SelItemIndex = 0 Then Return (EnteredNumber - 32) * (10 / 18)
        If ComBox3SelItemIndex = 1 Then Return EnteredNumber * 1
        If ComBox3SelItemIndex = 2 Then Return ((EnteredNumber - 32) * (5 / 9)) + 273.15
        Return 0
    End Function
    'This function returns Temperature, kelvin to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueKelvinTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        If ComBox3SelItemIndex = 0 Then Return (EnteredNumber - 273.15)
        If ComBox3SelItemIndex = 1 Then Return (EnteredNumber * 9 / 5) - 459.67
        If ComBox3SelItemIndex = 2 Then Return EnteredNumber * 1
        Return 0
    End Function
    'This function returns Weight, kg to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueKgTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 3
            If ForCounter = ComBox3SelItemIndex Then
                Return WeightRelation15(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
    'This function returns Weight, sher to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueSherTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 3
            If ForCounter = ComBox3SelItemIndex Then
                Return WeightRelation16(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function

    'This function returns Weight, gram to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueGramTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 3
            If ForCounter = ComBox3SelItemIndex Then
                Return WeightRelation17(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    'This function returns Weight, vori to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueVoriTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 3
            If ForCounter = ComBox3SelItemIndex Then
                Return WeightRelation18(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
    'This function returns Survey, acre to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueAcreTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 2
            If ForCounter = ComBox3SelItemIndex Then
                Return SurveyRelation19(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
    'This function returns Survey, decimal to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueDecimalTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 2
            If ForCounter = ComBox3SelItemIndex Then
                Return SurveyRelation20(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
    'This function returns Survey, Sqft to all combination calculated value. 3 parameters comes from calling sub
    Public Function CalculatedValueSurveySqFtTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
        Dim ForCounter As Integer
        For ForCounter = 0 To 2
            If ForCounter = ComBox3SelItemIndex Then
                Return SurveyRelation21(ForCounter) * EnteredNumber
            End If
        Next
        Return 0
    End Function
End Module
```

এখানেই শেষ হলো `Public Module CalculationModule` -এর কোডগুলো। আমরা এবার কোডের ওপর বিস্তারিত ব্যাখ্যা তুলে ধরার প্রয়াস পাচ্ছি। এ ব্যাখ্যা শেষ হওয়ার সাথে একক পরিবর্তন প্রোগ্রাম সম্পর্কে তথ্য প্রদানের কাজটিরও ইতি ঘটবে। শিক্ষার্থীদের প্রতি পুনরায় আবদার জানাচ্ছি, পুরো প্রোগ্রাম সম্পর্কে যা কিছু আলোচনা হয়েছে এবং যেসব কোড তুলে ধরেছি তা ভালো করে বুঝার চেষ্টা করবেন। প্রোগ্রামের লজিক ও কোড সম্পর্কে যতো বেশি বুঝ আসবে, ততো বেশি লাভ হবে। এবার ব্যাখ্যার পালা।

প্রথমেই বলে রাখি, মডিউলের ভেতর মোট ২৪টি ছোট ফাংশন ডিফাইন করা হয়েছে। এর কারণ হলো, আমাদের প্রোগ্রাম মোট ২৪টি একক পরিবর্তনে সক্ষম। এগুলো হলো: ১. দূরত্ব ক্যাটাগরির ৭টি [ক. মাইল, খ. কিলোমিটার, গ. মিটার, ঘ. মিলিমিটার, ঙ. গজ, চ. ফুট ও ছ. ইঞ্চি]। ২. বর্গ ক্যাটাগরির ৭টি [ক. বর্গমাইল, খ. বর্গকিলোমিটার, গ. বর্গমিটার, ঘ. বর্গমিলিমিটার, ঙ. বর্গগজ, চ. বর্গফুট ও ছ. বর্গইঞ্চি]। ৩. তাপ ক্যাটাগরির ৩টি [ক. সেন্টিগ্রেড, খ. ফারেনহাইট ও গ. কেলভিন]। ৪. ওজন ক্যাটাগরির ৪টি [ক. কিলোগ্রাম, খ. সের, গ. গ্রাম ও ঘ. ভরি]। ৫. জমিরা মাপ ক্যাটাগরির ৩টি [ক. একর, খ. শতাংশ ও গ. বর্গফুট]।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

মডিউল ডিক্লারেশনের পর ও কোনো ফাংশন শুরুর আগেই আমরা বেশ কিছু গ্লোবাল ভেরিয়েবল ডিক্লায়ার করেছি। আমাদের ইচ্ছে এগুলো যে কোনো ফাংশনে ব্যবহার করা। লক্ষ করুন, সব ভেরিয়েবল হলো এ্যারে। এগুলোর টাইপও দশমিক সংখ্যা [Double]। চারটি ব্লকে এগুলো ডিক্লায়ার করেছি সহজে সনাক্তকরণের জন্য। যেমন: ব্লক ১. দূরত্ব সম্পর্কিত এ্যারে। ব্লক ২. বর্গ সম্পর্কিত এ্যারে। ব্লক ৩. ওজন সম্পর্কিত এ্যারে। ব্লক ৪. জমির মাপ সম্পর্কিত এ্যারে। লক্ষণীয়, তাপ সম্পর্কে কোনো এ্যারে ভেরিয়েবলের প্রয়োজন নেই। এর কারণ হলো, তাপমাত্রার পরিবর্তন হয় বিশেষ ফর্মুলা দ্বারা। কিন্তু বাকি সবই দু'টি সংখ্যার গুণফল।

গ্লোবাল এ্যারে ডিক্লারেশনের পর আমরা একে একে ২৪টি ফাংশন ডিফাইন করেছি। এ প্রসঙ্গে প্রথম ফাংশনটি কিভাবে কাজ করে তা বুঝিয়ে বলবো। এরপর তাপমাত্রা ক্যালকুলেশনের ফাংশন সম্পর্কে বলবো। কারণ এগুলো একটু ভিন্ন। আশাকরি শিক্ষার্থীদের জন্য এটুকু ব্যাখ্যাই যথেষ্ট হবে। প্রথমে একটি ফাংশন কোড আবার দেখে নিই:

```
'These functions returns Lengh Catagory to all combination calculated value. 2 parameters comes from calling sub
Public Function CalculatedValueMileTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    Dim ForCounter As Integer
    For ForCounter = 0 To 6
      If ForCounter = ComBox3SelItemIndex Then
        Return LengthRelation1(ForCounter) * EnteredNumber
      End If
    Next
    Return 0
End Function
```

ব্যাখ্যা: প্রথমে জেনে রাখা দরকার যে, উক্ত ফাংশনটি কল করা হয়েছে ফর্মের ক্লাসের ভেতর ডিফাইন করা আরেকটি সাবরুটিন থেকে। সেখানে এই কোডটি দ্বারা ডাকা হয়েছে: `CalculatedValueMileTo(EnteredNumber, ComBox3SelItemIndex)`। লক্ষ করুন, কলিং কোডে ফাংশনের নাম ও প্যারামিটার একই। সুতরাং প্রোগ্রাম এই কোড ব্লক থেকে লাফ মেরে মডিউলের কোডে ডিফাইন করা একই নামের উক্ত ফাংশনে আসবে। অর্থাৎ উপরের ফাংশনটি এক্সেকিউট হবে। কিভাবে হবে? এক্ষুণি জেনে নিবেন।

প্রথম লাইনে আমরা একটি ইন্টেজার ভেরিয়েবল ডিক্লায়ার করেছি For-Next লুপটি চালানোর জন্য। এরপর সে লুপের ভেতর যেয়ে প্রয়োজনে মোট ৭বার পর্যন্ত [০-৬] ঘুরে দ্বিতীয় প্যারামিটর তথা `ComBox3SelItemIndex`-এ দেওয়া ইনডেক্স সংখ্যাটি খুঁজে বের করেছি। If ForCounter = ComBox3SelItemIndex কোডটি দ্বারা এ সন্ধান চালিয়েছি। কারণ প্রতিবার লুপ চলাকালে ForCounter -এর মান [০ থেকে ৬ পর্যন্ত] এক এক করে বাড়বে। আর দ্বিতীয় প্যারামিটামের মান অবশ্যই ০ থেকে ৬ পর্যন্ত যে কোনো একটি সংখ্যা হবেই। উভয় সংখ্যা সমান হলেই আমরা বুঝতে পারলাম কোন্ হিসাবটি দরকার। সুতরাং Return LengthRelation1(ForCounter) * EnteredNumber-দ্বারা সঠিক ফলাফল পাঠিয়ে দিচ্ছি কলিং সাবরুটিনে। এখানে লক্ষণীয়, হিসাব করতে দু'টি সংখ্যার প্রয়োজন। এর প্রথমটি আসবে গ্লোবেল ভেরিয়েবল দূরত্ব সম্পর্ক LengthRelation1 -থেকে। ভেরিয়েবলের এ্যারেতে মোট ৭টি সংখ্যা আছে। সঠিকটি সনাক্ত করে আনার জন্য ইনডেক্স প্রয়োজন। এই ইনডেক্সই হলো দ্বিতীয় প্যারামিটারের সমান [লুপ থেকে প্রাপ্ত] ForCounter এর মানটি। দ্বিতীয় সংখ্যাটি আসবে প্রথম প্যারামিটার থেকে- যা কলিং সাবই নির্ধারণ করে দিয়েছে। অর্থাৎ যে সংখ্যাটি ব্যবহারকারী টেক্সট ফিল্ডে ঢুকিয়েছেন সেটাই হলো EnteredNumber-এর মান। যদি কোনো কারণে সব ফেল করে তাহলে ফাংশন ০ রিটার্ন করবে। বাস্তবে সঠিকভাবে সংখ্যা ঢুকালে ফাংশন থেকে সর্বদাই সঠিক হিসাব রিটার্ন হবে। আগেই বলেছি বাকি প্রায় সবগুলো ফাংশনই একইভাবে কাজ করে। সুতরাং ওগুলো সম্পর্কে আর ব্যাখ্যার প্রয়োজন নেই। শিক্ষার্থীরা কিন্তু সব কোড দেখে নিবেন ও কিভাবে কাজ করে বুঝার চেষ্টা করবেন। শেষের দিকে ডিফাইন করা তাপমাত্রার ফাংশন সম্পর্কে কিছু ব্যাখ্যার প্রয়োজন আছে। সুতরাং পরের পৃষ্ঠায় আমরা এ সম্পর্কে তথ্যাদি তুলে ধরার প্রয়াস পাচ্ছি।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

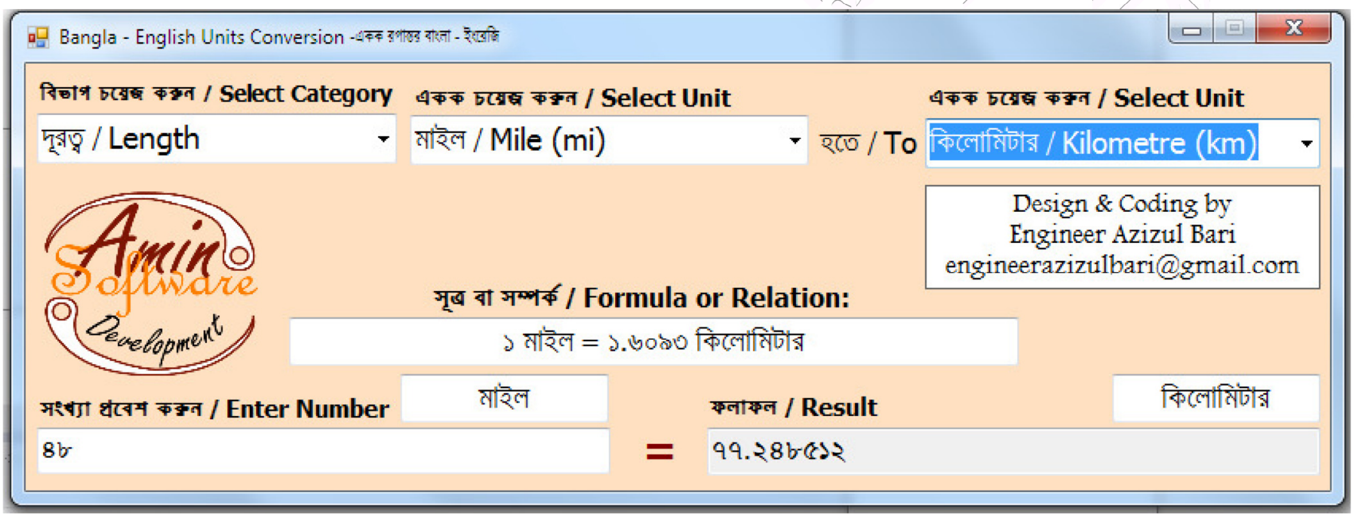
```
'This function returns Temperature, celcius to all combination calculated value. 2 parameters comes from calling sub
Public Function CalculatedValueCelciusTo(EnteredNumber As Double, ComBox3SelItemIndex As Integer) As Double
    If ComBox3SelItemIndex = 0 Then Return EnteredNumber * 1
    If ComBox3SelItemIndex = 1 Then Return (EnteredNumber * 1.8) + 32
    If ComBox3SelItemIndex = 2 Then Return EnteredNumber + 273.15
    Return 0
End Function
```

উপরে তাপ ক্যাটাগরির প্রথম ফাংশন কোড আবার তুলে ধরেছি। ফাংশনের ভেতর ৪টি মাত্র লাইন আছে। প্রথম তিনটি দ্বারাই আমরা সঠিক হিসাব করে কলিং সাবে ফলাফল ফেরৎ পাঠিয়েছি। যেমন: লাইন ১: এতে টেস্ট করেছি দ্বিতীয় প্যারামিটার এর মান ০ কি না? যদি হয়ে থাকে তাহলে ১ এর সঙ্গে প্রথম প্যারামিটার পূরণ হবে। অনুরূপ লাইন ২: দ্বিতীয় প্যারামিটারের মান ১ থাকলে সংখ্যা নির্ণিত হবে দেখানো ফর্মুলা মুতাবিক। লাইন ৩: দ্বিতীয় প্যারামিটারের মান ২ হলে প্রথম প্যারামিটারের সাথে ২৭৩.১৫ যোগ হবে। লক্ষ্য করুন, তিনটির মধ্যে শুধুমাত্র একটি মান সত্য। সুতরাং ফাংশন থেকে সত্য মানের সংখ্যা রিটার্ন হবে প্রতিবারই। আর কোনো কারণে সমস্যা হলো শেষের লাইনটি এক্সেকিউট হবে।

এ হচ্ছে 'Bangla / English Unit Conversion Software - বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার' - নামক আমাদের সৃষ্ট নতুন পিসি এ্যাপলিকেশনের কোডসহ পূর্ণাঙ্গ ব্যাখ্যা। কোডগুলো পেতে আমার নিকট ই-মেইল করুন এই ঠিকানায়: **engineerazizulbari@gmail.com**। আপনার ই-মেইলে কোড পাঠিয়ে দেবো। ফলে সময় ও কষ্টের বিনিময়ে এতো কোড নিজে টাইপ করা থেকে বিরত থাকতে পারেন। এছাড়া থাকতে পারেন ভুল-ত্রুটি থেকে নিরাপদ।

টেস্ট ও বিস্তারণ

বহু কষ্ট, সময় ক্ষেপণ ও ধৈর্যের বিনিময়ে আমরা তৈরি করেছি 'Bangla / English Unit Conversion Software - বাংলা / ইংরেজি একক পরিবর্তন সফটওয়্যার' নামক এ প্রোগ্রামটি। কোডিংয়ের সময় আপনাকে বার বার রান করে দেখতে হবে সবকিছু সঠিকভাবে কাজ করছে কি না। তবে যেহেতু এখন পুরো প্রোগ্রামের কাজ শেষ হয়ে গেছে তাই ফাইন্যাল টেস্ট করা জরুরী। আমরা চাই সকলেই এই প্রোগ্রাম থেকে উপকৃত হবে। সুতরাং ইন্টারনেট বা অন্য কোনো উপায়ে প্রোগ্রামটি পাবলিশ বা প্রকাশ করতে হবে। এ কাজের নামই বিস্তারণ। প্রথমে নিচের ছবিটি দেখুন।



উপরের চিত্রে আমরা মাইল থেকে কিলোমিটার একক পরিবর্তন করেছি। ৪৮ মাইলে হয় ৭৭.৬০৯৩ কি.মি। আমরা সব ফাংশন পরীক্ষা করে দেখেছি। প্রোগ্রাম সঠিক ফলাফল দিয়েছে। কোনো সমস্যা পরিলক্ষিত হয় নি। সুতরাং আমরা সফল! এবার এটি কিভাবে বিস্তারণ করবো তা জেনে নিই। কেমন?

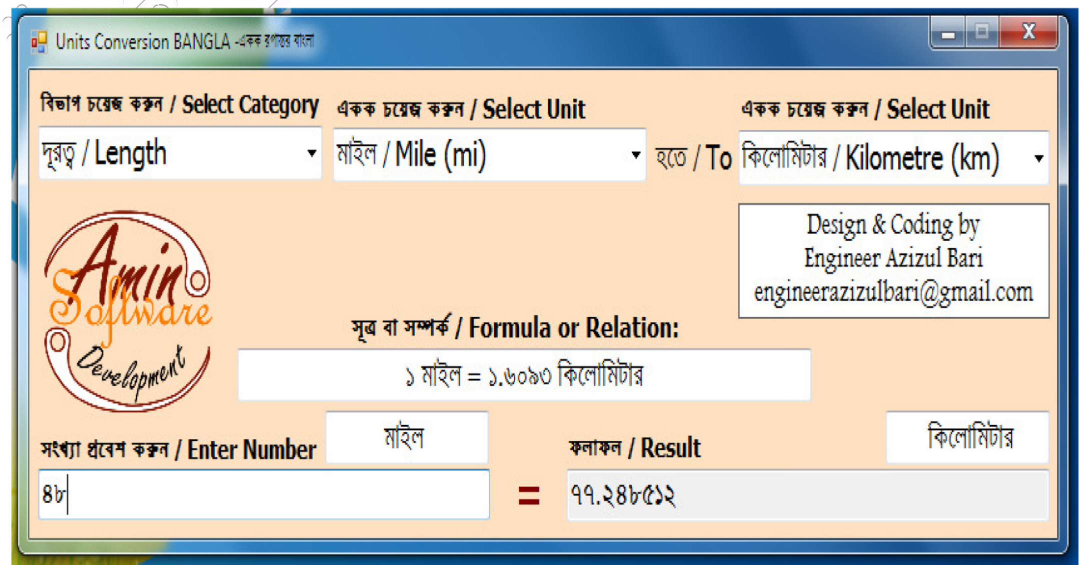
VB.NET

ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং

নতুন শিক্ষার্থীদের জন্য গুরুত্বপূর্ণ প্রশিক্ষণ রিসোর্স

দ্বিতীয় খণ্ড

Part Two



বিশেষ ইন্টারনেট সংস্করণ

Special Internet Edition

ইঞ্জিনিয়ার আজিজুল বারী
engineerazizulbari@gmail.com

খানকায়ে আমীনিয়া-আসগরিয়া গবেষণা বিভাগ
Ali Centre, Subidbazar, Sylhet, BD

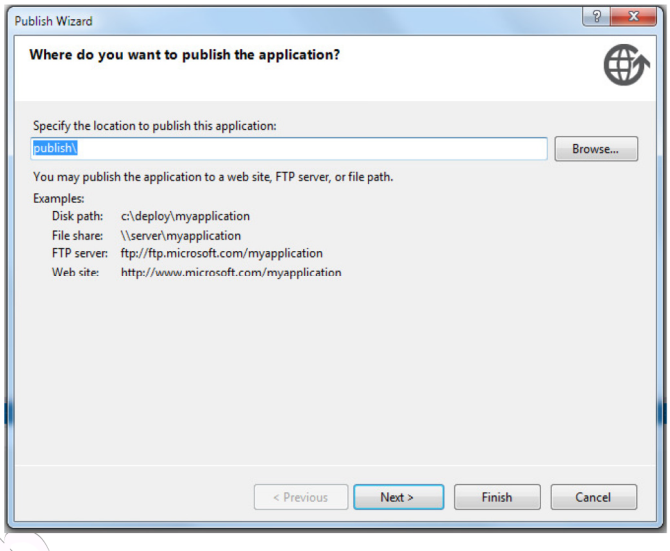
১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

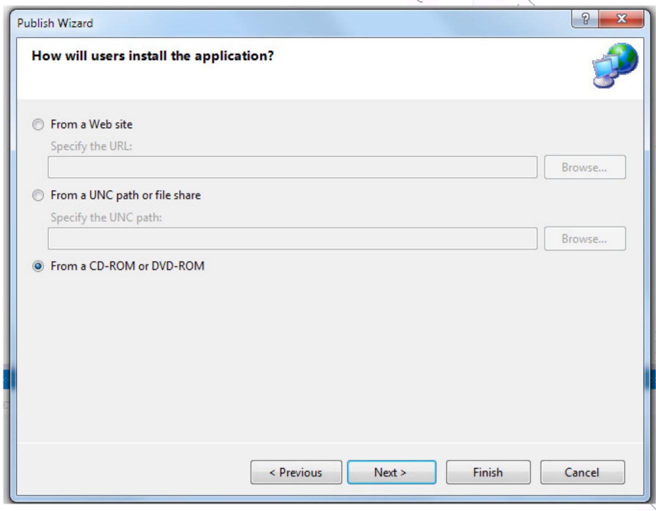
বিস্তারণ বা প্রকাশ করতে নিচের পদক্ষেপগুলো ধারাবাহিকভাবে অনুসরণ করুন:

১. বিল্ড [Build] মেনুতে যেয়ে চয়েজ করুন 'পাবলিশ ইউনিট্স কনভার্শন বাংলা'। নিচের উইজার্ড জানালাটি আসবে।

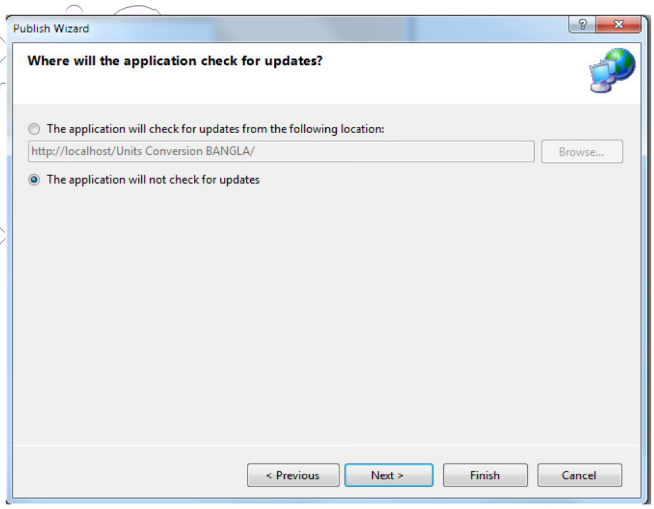
২. ব্রাউজ বোতামে ক্লিক করে কোথায় এটি পাবলিশ করবেন জানিয়ে দিন। নিচে কিছু উদাহরণ দেওয়া আছে। সুতরাং আমার উইজার্ডে দেখাবে এই প্যাথটি:
C:\Users\htci\Desktop\Bangla Units Conversion

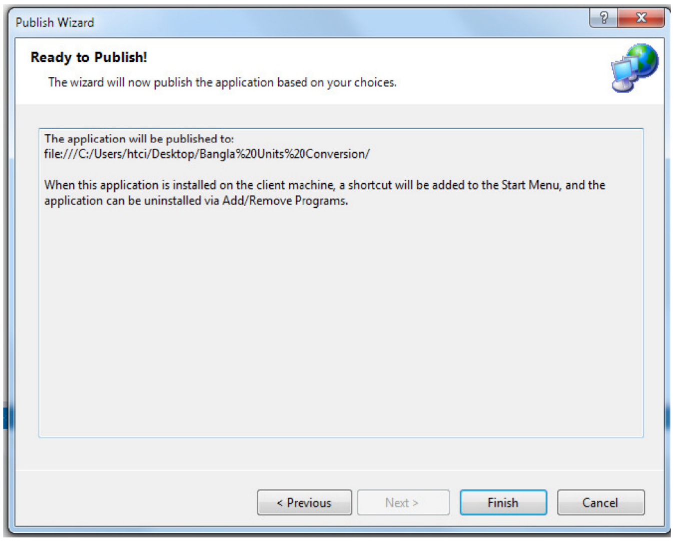


৩. এবার নেক্সট্ বোতাকে ক্লিক করুন। উইজার্ড এবার নিচের জানালাটি প্রতর্শন করবে। সর্বশেষ রেডিও বোতাম চয়েজ করা থাকবে। সম্ভবত আপনি এ চয়েজই করবেন।



এরপর নেক্সট্ বোতামে টিপ দিন। নিচের জানালা আসবে। এ্যাপলিকেশন আপডেইট কোথায় করবে? এ প্রশ্নের জবাবে আপনি চয়েজ করবেন নিচের রেডিও বোতাম। অর্থাৎ আপডেইট করবে না। এবার পুনরায় নেক্সট্ বোতামে ক্লিক করুন। নিচের ডায়লগটি আসবে। আপনার এ্যাপলিকেশন





পাবলিশের জন্য প্রস্তুত হয়ে গেছে! জানালায় কিছু তথ্য থাকবে: The application will be published to: file:///C:/Users/htci/Desktop/Bangla%20Units%20Conversion/
When this application is installed on the client machine, a shortcut will be added to the Start Menu, and the application can be uninstalled via Add/Remove Programs.। দেখতেই পাচ্ছেন- এতে আছে ফাইল প্যাথ। আরো আছে ইনস্টল করার পর স্টার্ট মেনুতে যে শর্টকাট আইকন থাকবে সে তথ্য।

৪. এবার ফিনিশ [Finish] বোতামে টিপ দিন। আপনার এ্যাপলিকেশন পাবলিশ হয়ে যাবে। ব্যস্। বন্ধুদেরকে এটি দেখিয়ে বাহাদুরী দেখাবার পালা এখন!

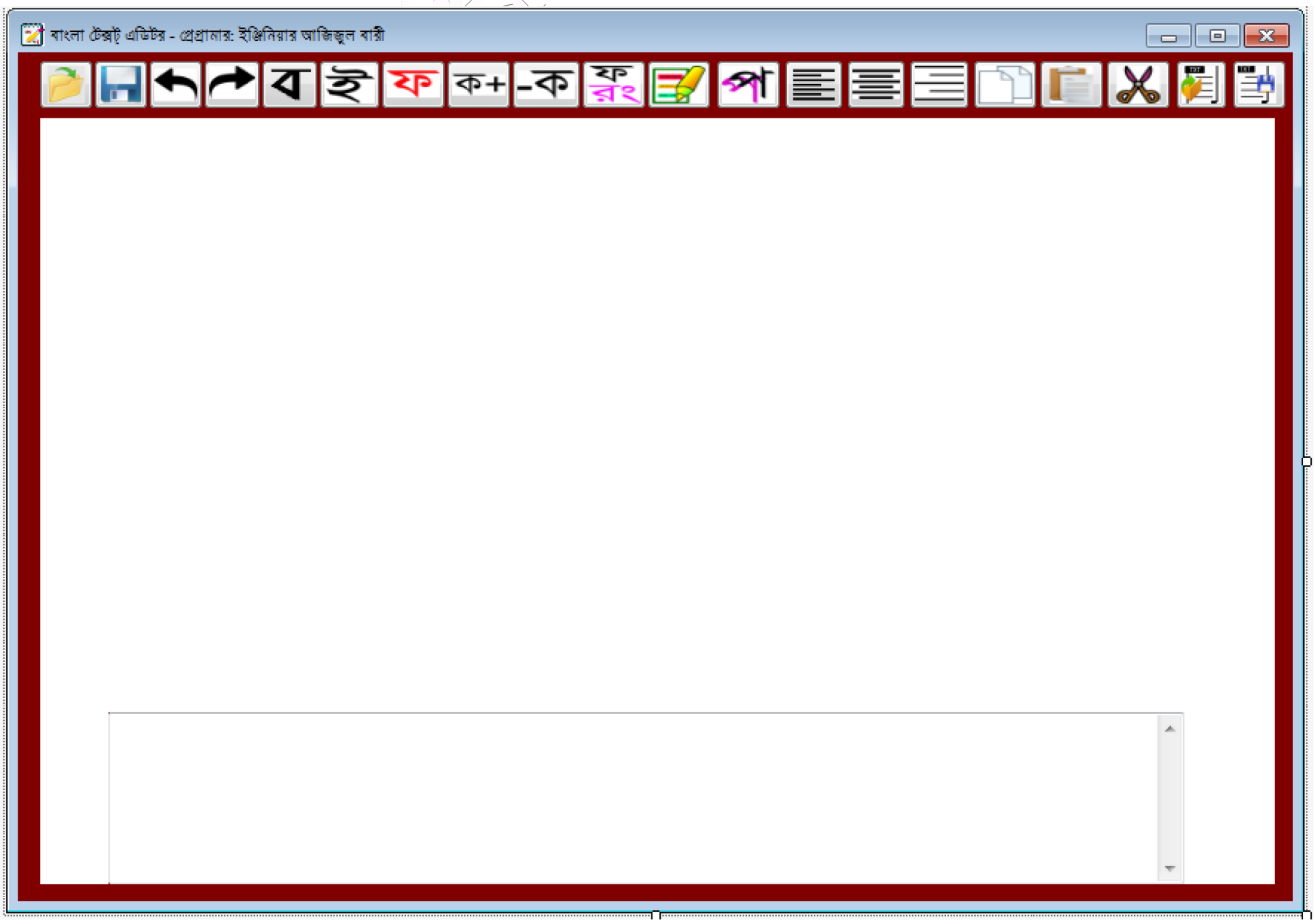
১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

Application 4 : Bangla / English Word Processor - বাংলা / ইংরেজি ওয়ার্ড প্রসেসর।

হ্যাঁ, ভিজুয়্যাল বেসিক.কম দ্বারা ওয়ার্ড প্রসেসর তৈরি করা যায়- আর তা আসলে খুব কঠিন কিছু নয়! অসংখ্য ওয়ার্ড প্রসেসর থাকাসত্ত্বেও আরেকটি কেনো? জবাবে বলা যায়, কেনো নয়? আমরা তো শিখবো- ঠিক না? তাছাড়া বাংলার ওপর গুরুত্ব দিয়ে এটি আমরা তৈরি করতে চাই। সুতরাং আসুন, গড়ে তুলি আমাদের চতুর্থ এ প্রোগ্রামটি- বাংলা-ইংরেজি ওয়ার্ড প্রসেসর।

১. ইউআই জিডাইন: প্রথমেই আমরা ডিজাইন করবো একটি আকর্ষণীয় ইউজার ইন্টারফেইস। নিচের চিত্রে একটি স্যাম্পুল দেখিয়েছি। শিক্ষার্থীরা চাইলে অন্যভাবেও ডিজাইন করতে পারেন। বরং চেষ্টা করবেন, নিজের মনের মতো ডিজাইন করতে। তবে মনে রাখবেন, নিম্নে তালিকাভুক্ত বস্তুগুলো ফর্মে থাকতে হবে।



(ক) একটি উইন্ডোজ ফর্ম:

(১) টেক্সট্ : “বাংলা টেক্সট্ এডিটর” [আমি নিজের নামটিও লিখেছি। আপনি আপনার নাম লিখতে পারেন]। (২) অটোস্কেলমোড [AutoScaleMode] = “Font”। (৩) পেছনের রং = Maroon। (৪) ফন্ট = “Times New Roman, 14pt]”। (৫) সম্ভব হলো একটি ‘প্রোগ্রাম’ আইকোন। [আমি একটি ও বাকী সব আইকোন নিজে ডিজাইন করে দিয়েছি। সব আইকোন ছবি প্রজেক্টে এ্যামবেড [যুক্ত] করেছি। কিভাবে করা যায়- তা একটু পরই লিখে দেবো। (৬) ফর্মের সাইজ = ৮৭৬, ৬০৮। ব্যস! বাকি সব প্রোপ্রাটি যেভাবে আছে সেভাবেই থাকবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

(খ) ২০টি ছোট্ট সাইজের বোতাম [বাম থেকে]: (১) OpenFileBt, (২) SaveFileBt, (৩) UndoBt, (৪) RedoBt, (৫) BoldBt, (৬) ItalicBt, (৭) FontBT, (৮) SizeUpBt, (৯) SizeDnBt, (১০) FontColBt, (১১) HighLiteColBt (১২) BckColBt, (১৩) AlinLBt, (১৪) AlinCBt, (১৫) AlinRBt, (১৬) CopyBt, (১৭) PstBt, (১৮) CutBt, (১৯) OpenTxtBt, (২০) SaveTxtBt। এগুলো সবই বোতামের নাম (name) প্রোপার্টি হবে। কোডে আমরা এসব নাম দ্বারাই বোতামগুলোকে সম্বোধন করবো। আপনি চাইলে আইকোন যুক্ত করতে পারেন। তবে সংক্ষেপে বাংলা বা ইংরেজি Text প্রেপার্টিও লিখতে পারেন। আইকোন দিলে বোতামের BackColour হবে ট্রান্সপ্যারেন্ট। এর পরের প্রেপার্টি তথা BackgroundImage প্রোপার্টিতে আইকোনের ছবি দেবেন। আর প্রতিটি বোতামের ফন্ট হবে Times New Roman, 14pt।

(গ) একটি RichTextBox1 কন্ট্রোল। টুলবাক্স চয়েজ করে অলউইন্ডোজ ফর্মস্ ট্যাবে ক্লিক করলেই নিচে দেখতে পাবেন এই কন্ট্রোলটি। এই কন্ট্রোলের বৈশিষ্ট্য হলো: একই সময় বিভিন্ন ফন্ট, সাইজ, রঙ ইত্যাদি ফরমাটে লেখা যায়। সুতরাং আমরা এতে বাংলা ও ইংরেজি লিখতে পারবো অনায়াসেই।

(ঙ) একটি লুকানো টেক্স বাক্স: (১) এর ফন্ট প্রোপার্টি হবে TonnyBanglaMJ, 14pt [বা বিজয়ের অন্য কোনো ফন্ট হলেও চলবে]। (২) এর মাল্টিলাইন প্রোপার্টি হবে True। (৩) স্ক্রলবার্স প্রোপার্টি হবে Vertical।

ব্যস! আর কিছু নেই। একবার প্রোগ্রাম রান করে দেখে নিন। আপনি চাইলে টেক্সট বাক্সে বাংলা বা ইংরেজিতে লিখতেও পারবেন। তবে আমরা তো চাই আরো কিছু ফাংশন্যালিটি! তাই আসুন প্রতিটি বোতামের কোড কি হবে দেখে নিই। প্রথমে পুরো কোডই একসাথে তুলে ধরছি। এরপর বিভিন্ন সাবরুটিনের ব্যাখ্যা একে একে তুলে ধরবো।

প্রোগ্রামের কোড:

```
Public Class Form1
    'কোডিং: ইঞ্জিনিয়ার আজিজুল বারী: engineerazizulbari@gmail.com
    Dim RichTextBoxSize As Integer = 838 'এই ইন্টেজার গ্লেবাল দ্বারা টেক্সট বাক্সের সাইজ নিয়ন্ত্রণ করা হবে।
    'ফর্ম লোড ইভেন্ট সাবের কোড।
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        'টেক্সট বাক্সের সাইজ সেট করা।
        RichTextBox1.Size = New Point(RichTextBoxSize, RichTextBox1.Height)
        'ফর্মের মাঝখানে টেক্সট বাক্স নিয়ে আসা।
        RichTextBox1.Location = New Point((Me.Size.Width - RichTextBoxSize) / 2, RichTextBox1.Location.Y)
        'কন্টেক্সট মেনু দেখানোর জন্য নিচে ডিফাইন করা সাবটি কল করা। টেক্সট বাক্সের ভেতর রাইট-ক্লিক করলে এই মেনু আসবে।
        InitializeMyContextMenu() 'নিচের সাবরুটিন কল।
    End Sub
    'এই সাবই কনটেক্সট মেনু প্রদর্শন করবে যখনই ব্যবহারকারী রাইট-ক্লিক করবেন টেক্সট বাক্সের ভেতর।
    Private Sub InitializeMyContextMenu()
            'কনটেক্সট মেনু ও মেনু আইটেম সৃষ্টি করা।
        Dim contextMenu1 As New ContextMenu() 'প্রথমে একটি নতুন কনটেক্সট মেনু ডিফাইন করতে হবে।
        Dim menuItem1 As New MenuItem("&কাটুন") 'এবার আরেকটি নতুন মেনু আইটেম ডিফাইন করে তাতে লেখা দিতে হবে।
        AddHandler menuItem1.Click, AddressOf CutBt_Click 'কোন বোমাত তা হ্যান্ডল করবে, জানাতে হবে।
        Dim menuItem2 As New MenuItem("&কপি")
            AddHandler menuItem2.Click, AddressOf CopyBt_Click
             Dim menuItem3 As New MenuItem("&পেস্ট")
         AddHandler menuItem3.Click, AddressOf PstBt_Click
          'মেনু আইটেমস্ প্রোপার্টি ব্যবহার করে এড মেথড ডাকতে হবে।
        'ফলে মেনু আইটেমগুলো মূল মেনু আইটেম কালেকশনে সংযুক্ত হবে।
      contextMenu1.MenuItems.Add(menuItem1)
      contextMenu1.MenuItems.Add(menuItem2)
      contextMenu1.MenuItems.Add(menuItem3)
      'mainMenu1-কে রিচটেক্সটবাক্সের সঙ্গে এসাইন করতে হবে।
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        RichTextBox1.ContextMenu = contextMenu1
    End Sub
        'ফর্মের রিসাইজ ইভেন্ট সাবের কোড। ফর্মের সাইজ পরিবর্তন হলে রিচ টেক্সট বাক্সের সাইজ ও পরিবর্তন হতে হবে।
    Private Sub Create_Resize(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Resize
        If Me.Size.Width > RichTextBoxSize + 50 Then 'যদি ফর্মের প্রস্থ সাইজ হয় বাক্সের সাইজ + ৫০ পয়েন্ট থেকে বেশি-
            RichTextBox1.Width = Me.Size.Width - 100 'তাহলে বাক্সের প্রস্থ সাইজ হবে একশ পয়েন্ট কম।
            RichTextBox1.Anchor = AnchorStyles.None 'কন্টেইনারের ভেতর কোনো বিশেষ স্টাইলে টেক্সট বাক্স থাকবে না।
            RichTextBox1.Anchor = AnchorStyles.Top Or AnchorStyles.Bottom 'শুধু উপর ও নিচের পার্শ্বের নিকটে থাকবে।
            'নিচের লাইনটির কাজ হলো ফর্মের সেন্টারে টেক্স বাক্স থাকার নিশ্চয়তা দেওয়া- সর্বদা।
            RichTextBox1.Location = New Point((Me.Size.Width - (Me.Size.Width - 100)) / 2, RichTextBox1.Location.Y)
        End If
        'নিচের কোড দ্বারা ফর্মের ভেতর টেক্সড্ বাক্সের অবস্থান, প্রস্থ ও আনকর নিয়ন্ত্রণ করা হয়েছে।
            If Me.Size.Width <= RichTextBoxSize + 50 Then
          If RichTextBox1.Location.X < 10 Then
              RichTextBox1.Location = New Point(5, RichTextBox1.Location.Y)
          End If
          If RichTextBox1.Width + 25 > Me.Size.Width Then
              RichTextBox1.Size = New Point(Me.Size.Width - 10 - RichTextBox1.Location.X, RichTextBox1.Size.Height)
          End If
          RichTextBox1.Anchor = AnchorStyles.None
          RichTextBox1.Anchor = AnchorStyles.Top Or AnchorStyles.Bottom Or AnchorStyles.Left Or AnchorStyles.Right
        End If
    End Sub
    'এটা আনডু বোতামের কোড।
   Private Sub UndoBt_Click(sender As Object, e As EventArgs) Handles UndoBt.Click
        RichTextBox1.Undo() 'যা এই মাত্র করা হয়েছে তা আনডু করবে।
         RichTextBox1.Focus()
    End Sub
    'এটা রিডু বোতামের কোড।
  Private Sub RedoBt_Click(sender As Object, e As EventArgs) Handles RedoBt.Click
         RichTextBox1.Redo() 'যা এই মাত্র আনডু হয়েছে তা রিডু করবে।
         RichTextBox1.Focus()
    End Sub
    'এটা বল্ড তৈরির কোড।
    Private Sub BoldBt_Click(sender As Object, e As EventArgs) Handles BoldBt.Click
        If RichTextBox1.SelectionFont.Bold = True Then 'যদি বল্ড থেকে তাকে তাহলে নরম্যাল বানাতে হবে।
            If RichTextBox1.SelectionFont.Italic = True Then 'ইটালিক থাকলে সেটা সহ নরম্যাল করতে হবে- যা নিচের কোড লাইনের কাজ।
                RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Regular +
FontStyle.Italic)
            Else 'অন্যথায় ইটালিক ছাড়াই একই কাজ করতে হবে।
                RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Regular)
            End If
        ElseIf RichTextBox1.SelectionFont.Bold = False Then 'যদি বল্ড না থাকে তাহলে বল্ড করতে হবে।
            If RichTextBox1.SelectionFont.Italic = True Then 'ইটালিক থাকলে সেটা সহ বল্ড করতে হবে- যা নিচের কোড লাইনের কাজ।
                RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Bold + FontStyle.Italic)
            Else 'অন্যথায় ইটালিক ছাড়াই একই কাজ করতে হবে।
                RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Bold)
            End If
        End If
        RichTextBox1.Focus() 'ফেকাস হবে টেক্সট বাক্সে।
    End Sub
    'এটা হলো ইটালিক বোতামের কোড।
Private Sub ItalicBt_Click(sender As Object, e As EventArgs) Handles ItalicBt.Click
        If RichTextBox1.SelectionFont.Italic = True Then 'ইটালিক থাকলে নরম্যাল হবে।
            If RichTextBox1.SelectionFont.Bold = True Then 'বল্ড আছে কি? থাকলে নিচের লাইন নিশ্চিত করবে বল্ডসহ নরম্যাল করা।
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
            RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Regular +
FontStyle.Bold)
         Else 'অন্যথায় এমনিতেই নরম্যাল হবে। নিচের লাইন এর নিশ্চয়তা দেবে।
            RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Regular)
         End If
      ElseIf RichTextBox1.SelectionFont.Italic = False Then 'ইটালিক না থাকলে ইলালিক হবে।
         If RichTextBox1.SelectionFont.Bold = True Then 'বল্ড আছে কি? থাকলে নিচের লাইন নিশ্চিত করবে বল্ডসহ ইটালিক করা।
            RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Italic + FontStyle.Bold)
         Else 'অন্যথায় এমনিতেই ইটালিক হবে। নিচের লাইন এর নিশ্চয়তা দেবে।
            RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Italic)
         End If
      End If
      RichTextBox1.Focus() 'ফেকাস হবে টেক্সট বাক্সে।
End Sub
   'ফন্ট পরিবর্তন বোতামের কোড।
Private Sub FontBt_Click(sender As Object, e As EventArgs) Handles FontBt.Click
      Dim fontDialod As New FontDialog 'প্রথম একটি নতুন ফন্ট ডায়লগ ডিফাইন করা চাই।
      fontDialod.Font = RichTextBox1.SelectionFont 'এখন সিলেক্ট করা ফন্ট কি জানতে হবে।
      fontDialod.ShowDialog() 'ফন্ট ডায়লগ প্রদর্শন।
      RichTextBox1.SelectionFont = fontDialod.Font 'সিলেক্ট করা ফন্ট পরিবর্তন।
      RichTextBox1.Focus() 'ফোকাস টেক্সট বাক্সে নিয়ে আসা।
End Sub
'ফট সাইজ বড়ো করার বোতামের কোড। এটা ঠিক নিচের সাবের কোডই- পার্থক্য হলে ৫ পয়েন্ট করে সাইজ বাড়বে।
Private Sub SizeUpBt_Click(sender As Object, e As EventArgs) Handles SizeUpBt.Click
      Try
         RichTextBox1.SelectionFont = New Font(RichTextBox1.SelectionFont.FontFamily, Int
    (RichTextBox1.SelectionFont.SizeInPoints + 5))
      Catch ex As Exception
      End Try
        RichTextBox1.Focus()
End Sub
    'ফট সাইজ ছোট করার বোতামের কোড।
Private Sub SizeDnBut_Click(sender As Object, e As EventArgs) Handles SizeDnBut.Click
      Try 'নিচের দীর্ঘ লাইনটি সিলেক্ট করা ফন্টের সাইজ কমাবে। লক্ষ করুর ৫ পয়েন্ট করে ফন্টের সাইজ কমবে প্রত্যেকবার ক্লিক করলেই।
        RichTextBox1.SelectionFont = New Font(RichTextBox1.SelectionFont.FontFamily, Int
        (RichTextBox1.SelectionFont.SizeInPoints - 5))
        Catch ex As Exception
      End Try
        RichTextBox1.Focus()
End Sub
'ফন্ট রং পরিবর্তন বোতামের কোড। এ কোড নিচের সাবের কোডের অনুরূপ।
Private Sub FontColBt_Click(sender As Object, e As EventArgs) Handles FontColBt.Click
    Dim colorDialog As New ColorDialog
    colorDialog.Color = RichTextBox1.SelectionColor
    colorDialog.ShowDialog()
    RichTextBox1.SelectionColor = colorDialog.Color
    RichTextBox1.Focus()
End Sub
'হাইলাইট রং পরিবর্তন বোতামের কোড।
Private Sub HighLiteColBt_Click(sender As Object, e As EventArgs) Handles HighLiteColBt.Click
    Dim colorDialog As New ColorDialog 'ডিফাইন নতুন একটি কালার ডায়লগ।
    'কোন্ রং আছে তা ডায়লগে দেওয়া।
    colorDialog.Color = RichTextBox1.SelectionBackColor
    colorDialog.ShowDialog() 'ডায়লগ প্রদর্শন
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    'চয়েজকরা রং হবে টেক্সট বাক্সের হাইলাইট রং। নিচের কোডের কাজ এটি।
    RichTextBox1.SelectionBackColor = colorDialog.Color
    RichTextBox1.Focus() 'টেক্সড় বাক্সকে ফোকাসে দাও।
End Sub
'পেছনের রং পরিবর্তন বোতামের কোড।
Private Sub BckColBt_Click(sender As Object, e As EventArgs) Handles BckColBt.Click
      Dim colorDialog As New ColorDialog 'ডিফাইন নতুন একটি কালার ডায়লগ।
      colorDialog.Color = RichTextBox1.BackColor 'কোন্ রং আছে তা ডায়লগে দেওয়া।
      colorDialog.ShowDialog() 'ডায়লগ প্রদর্শন
      RichTextBox1.BackColor = colorDialog.Color 'চয়েজকরা রং হবে টেক্সট বাক্সের পেছনের রং।
      RichTextBox1.Focus() 'টেক্সড় বাক্সকে ফোকাসে দাও।
End Sub
'বামদিকে এলাইনের বোতামের কোড।
Private Sub AlinLBt_Click(sender As Object, e As EventArgs) Handles AlinLBt.Click
      RichTextBox1.SelectionAlignment = HorizontalAlignment.Left
End Sub
'মাঝখানে এলাইনের বোতামের কোড।
Private Sub AlinCBt_Click(sender As Object, e As EventArgs) Handles AlinCBt.Click
      RichTextBox1.SelectionAlignment = HorizontalAlignment.Center
End Sub
 'ডানদিকে এলাইনের বোতামের কোড।
Private Sub AlnRBt_Click(sender As Object, e As EventArgs) Handles AlnRBt.Click
      RichTextBox1.SelectionAlignment = HorizontalAlignment.Right
End Sub
'কপি বোতামের কোড। লক্ষ করুন, নিচের কাট বোতামের কোড ও এই বোতামের কোড একই-
'তবে একটি লাইন এখানে নেই- মুছে দেওয়ার লাইনটি এখানে প্রযোজ্য নয়।
Private Sub CopyBt_Click(sender As Object, e As EventArgs) Handles CopyBt.Click
      My.Computer.Clipboard.Clear()
      Try
         Clipboard.SetText(RichTextBox1.SelectedText)
      Catch ex As Exception
      End Try
End Sub
'কাট বোতামের কোড।
Private Sub CutBt_Click(sender As Object, e As EventArgs) Handles CutBt.Click
      My.Computer.Clipboard.Clear() 'আগে ক্লিপবোর্ড পরিষ্কার করতে হবে।
      Try 'যাতে সমস্যা হলে প্রোগ্রাম ক্রাশ করে না- তাই এই ছোট্ট কথাটি।
         'প্রথমে ক্লিপর্বোর্ডে সিলেক্ট করা লেখা নিয়ে যেতে হবে- নিচের লাইন এ কাজই করে।
         Clipboard.SetText(RichTextBox1.SelectedText)
                RichTextBox1.SelectedText = "" 'এবার টেক্সট বাক্স থেকে মুছে দিতে হবে লেখাটুকু।
      Catch ex As Exception 'যদি ভুল হয়! ধরো!
      End Try
End Sub
'পেস্ট বোতামের কোড।
Private Sub PstBt_Click(sender As Object, e As EventArgs) Handles PstBt.Click
      If My.Computer.Clipboard.ContainsText Then 'প্রথমে দেখতে হবে ক্লিপবোর্ডে লেখা আছে।
         RichTextBox1.Paste() 'থাকলে ইনসারশন পয়েন্টে পেস্ট করার কোড।
   End If
End Sub
 'ফাইল খোলার কোড।
Private Sub OpenFileBt_Click(sender As Object, e As EventArgs) Handles OpenFileBt.Click
      'প্রথমে খোলার ডায়লগ খুলতে হবে।
      Dim openFile1 As New OpenFileDialog()
      'ফাইল আর.এফটি. টাইপ কোথায় আছে দেখতে হবে।
      openFile1.DefaultExt = "*.rtf"
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        openFile1.Filter = "RTF Files|*.rtf"
        'ফাইল খোলার ডায়লগ দ্বারা কি কোনো ফাইল খোলার নির্দেশ এসেছে?
        If (openFile1.ShowDialog() = System.Windows.Forms.DialogResult.OK) _
            And (openFile1.FileName.Length > 0) Then 'ফাইল নামটি এখন জানা।
          'এখন ফাইলের সবকিছু আমাদের টেক্সট বাক্সে নিয়ে আসতে হবে। এজন্য নিচের কোড লাইনটি।
            RichTextBox1.LoadFile(openFile1.FileName, RichTextBoxStreamType.RichText)
            Me.Text = openFile1.FileName 'এই কোডটুকু দ্বারা প্রদর্শিত ফাইলের প্যাথনেইম ফর্মের উপরে আইকনের পাশে আসবে।
      End If
End Sub
'ফাইল সেইভ করার কোড।
Private Sub SaveFileBt_Click(sender As Object, e As EventArgs) Handles SaveFileBt.Click
        'প্রথমে সেইভ ডায়লগ খুলতে হবে।
    Dim saveFile1 As New SaveFileDialog()
        'ফাইল আর.এফটি. টাইপ হওয়ার জন্য নির্দেশ থাকতে হবে।
        saveFile1.DefaultExt = "*.rtf"
        saveFile1.Filter = "RTF Files|*.rtf"
        'নিচের কোড নিশ্চিত করছে আপনি ফাইল সেইভ ডায়লগে সেইভ করার ইচ্ছা করেছেন। না করলে কিছুই হবে না।
      If (saveFile1.ShowDialog() = System.Windows.Forms.DialogResult.OK) _
          And (saveFile1.FileName.Length > 0) Then 'আপনি ফাইলেন নামও দিয়েছেন বলে নিশ্চিত হওয়া।
          'এবার ফাইল সংরক্ষণ হবে নিচের কোড লাইনটির মাধ্যমে।
            RichTextBox1.SaveFile(saveFile1.FileName, RichTextBoxStreamType.RichText)
      End If
End Sub
'প্রোগ্রাম বন্ধ হওয়ার সময় কিছু লেখা থাকলে এই সাব সক্রিয় হয়ে প্রশ্ন করবে সেইভ করবেন কি না?
Private Sub Form1_Closing(sender As Object, e As System.ComponentModel.CancelEventArgs) Handles
MyBase.Closing
        If RichTextBox1.Text <> "" Then 'বাক্সে লিখা আছে- তা নিশ্চিত করা চাই।
         'নিচের লাইন দ্বারা আপনাকে প্রশ্ন করা ও ফলাফল ভেরিয়েবলে সংরক্ষণ করা হবে।
         Dim Response As String = MsgBox("লেখা সংরক্ষণ করবেন?", vbYesNo)
            If Response = vbYes Then 'যদি ফাইল রক্ষা করতে চান।
                SaveFileBt.PerformClick() 'তাহলে এ সেইভ বোতামে ক্লিক করা হবে স্বয়ংক্রিয়ভাবে।
        End If
        End If
End Sub
 'এই ১৮টি সাবরুটিন হলো আঠারোটি বোতামের মাউস-প্রবেশ সিস্টেম ইভেন্ট সাব।
'আপনি যখন মাউস বোতামের ভেতর নিবেন তখনই বাংলায় কিছু নির্দেশনা দেখবেন।
Private Sub AlinCBt_MouseEnter(sender As Object, e As EventArgs) Handles AlinCBt.MouseEnter
     Dim s As String = "সেন্টার এলাইন" 'প্রথমে একটি স্ট্রিং ভেরিয়েবলে যা লেখা হবে তা সংরক্ষণ করেছি।
     'নিচের লাইনটি হলো  টুলটিপ  নামক সিস্টেম ফাংশন ব্যবহারের লাইন। এর পরেরটি দ্বারা প্রদর্শন হবে।
     Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
     tooltip1.SetToolTip(AlinCBt, s)
End Sub
Private Sub AlinLBt_MouseEnter(sender As Object, e As EventArgs) Handles AlinLBt.MouseEnter
        Dim s As String = "লেফট এলাইন"
        Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
        tooltip1.SetToolTip(AlinLBt, s)
End Sub
Private Sub AlnRBt_MouseEnter(sender As Object, e As EventArgs) Handles AlnRBt.MouseEnter
     Dim s As String = "রাইট এলাইন"
     Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
     tooltip1.SetToolTip(AlnRBt, s)
End Sub
Private Sub BckColBt_MouseEnter(sender As Object, e As EventArgs) Handles BckColBt.MouseEnter
     Dim s As String = "পেছনের রং"
     Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    tooltip1.SetToolTip(BckColBt, s)
End Sub
Private Sub BoldBt_MouseEnter(sender As Object, e As EventArgs) Handles BoldBt.MouseEnter
    Dim s As String = "বল্ড লেখা"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(BoldBt, s)
End Sub
Private Sub Button1_MouseEnter(sender As Object, e As EventArgs) Handles OpenFileBt.MouseEnter
    Dim s As String = "ফাইল খুলুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(OpenFileBt, s)
End Sub
Private Sub Button2_MouseEnter(sender As Object, e As EventArgs) Handles SaveFileBt.MouseEnter
    Dim s As String = "সংরক্ষণ করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(SaveFileBt, s)
End Sub
Private Sub CopyBt_MouseEnter(sender As Object, e As EventArgs) Handles CopyBt.MouseEnter
    Dim s As String = "কপি করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(CopyBt, s)
End Sub
Private Sub CutBt_MouseEnter(sender As Object, e As EventArgs) Handles CutBt.MouseEnter
    Dim s As String = "কেটে দিন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(CutBtn, s)
End Sub
Private Sub FontBt_MouseEnter(sender As Object, e As EventArgs) Handles FontBt.MouseEnter
    Dim s As String = "ফন্ট পরিবর্তন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(FontBt, s)
End Sub
Private Sub FontColBt_MouseEnter(sender As Object, e As EventArgs) Handles FontColBt.MouseEnter
    Dim s As String = "ফন্ট রং"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(FontColBt, s)
End Sub
Private Sub HighLiteColBt_MouseEnter(sender As Object, e As EventArgs) Handles HighLiteColBt.MouseEnter
    Dim s As String = "হাইলাইট রং"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(HighLiteColBt, s)
End Sub
Private Sub ItalicBt_MouseEnter(sender As Object, e As EventArgs) Handles ItalicBt.MouseEnter
    Dim s As String = "ইটালিক লেখা"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(ItalicBt, s)
End Sub
Private Sub PstBt_MouseEnter(sender As Object, e As EventArgs) Handles PstBt.MouseEnter
    Dim s As String = "পেস্ট করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(PstBt, s)
End Sub
Private Sub RedoBt_MouseEnter(sender As Object, e As EventArgs) Handles RedoBt.MouseEnter
    Dim s As String = "আবার করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    tooltip1.SetToolTip(RedoBt, s)
End Sub
Private Sub SizeDnBut_MouseEnter(sender As Object, e As EventArgs) Handles SizeDnBut.MouseEnter
    Dim s As String = "ছোট ফন্ট"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(SizeDnBut, s)
End Sub
Private Sub SizeUpBt_MouseEnter(sender As Object, e As EventArgs) Handles SizeUpBt.MouseEnter
    Dim s As String = "বড়ো ফন্ট"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(SizeUpBt, s)
End Sub
Private Sub UndoBt_MouseEnter(sender As Object, e As EventArgs) Handles UndoBt.MouseEnter
    Dim s As String = "আনডু করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(UndoBt, s)
End Sub
Private Sub OpenTxtBt_MouseEnter(sender As Object, e As EventArgs) Handles OpenTxtBt.MouseEnter
    Dim s As String = "টেক্সট ফাইল খুলুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(OpenTxtBt, s)
End Sub
Private Sub SaveTxtBt_MouseEnter(sender As Object, e As EventArgs) Handles SaveTxtBt.MouseEnter
    Dim s As String = "টেক্সট ফরমাটে সংরক্ষণ করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(SaveTxtBt, s)
End Sub
'টেক্সট ফাইল খোলার কোড।
Private Sub OpenTxtBt_Click(sender As Object, e As EventArgs) Handles OpenTxtBt.Click
    Dim Open As New OpenFileDialog() 'ফাইল খোলার ডায়লগ ক্লাসের নতুন ইন্সট্যান্ট।
    Dim myStreamReader As System.IO.StreamReader 'একটি সিস্টেম স্ট্রিমরিডার অবজেক্ট ডাইমেনশন করা।
    Open.Filter = "Text[*.txt*]|*.txt|All Files [*,*]|*.*" 'টেক্সট টাইপ ফিল্টার।
    Open.CheckFileExists = True 'খোলার পূর্বে নিশ্চিত করতে হবে ফাইলটি আছে কি না।
    Open.Title = "টেক্সট ফাইল খুলুন" 'ডায়লগের টাইটেল।
    Open.ShowDialog(Me) 'ডায়লগ প্রদর্শন
    Try 'এটা একটি ডিবাগ কিওয়ার্ড। সমস্যা হলে পরের চার লাইন ছেড়ে  ক্যাচ ইএক্স  লাইনে জাম্প দেবে।
        Open.OpenFile() 'চয়েজ করা ফাইল খোলার নির্দেশ।
        myStreamReader = System.IO.File.OpenText(Open.FileName)
        Dim NewRFTText As String = myStreamReader.ReadToEnd() 'ফাইল পুরোটা রিড করার নির্দেশ।
        RichTextBox1.Text = NewRFTText 'ভেরিয়েবলে সংরক্ষিত রিড-করা পুরো ফাইল রিচটেক্সট বাক্সে প্রদর্শন।
      Catch ex As Exception 'কোনো সমস্যা হলো ডায়লগ এসে প্রদর্শন করবে।
    End Try
End Sub
Private Sub SaveTxtBt_Click(sender As Object, e As EventArgs) Handles SaveTxtBt.Click
    If RichTextBox1.Text <> "" Then ' লেখা আছে কি না দেখেই নিচের সাবরুটির কল করতে হবে।
      SaveTheFile() 'নিচের সাবরুটিন কল।
    End If
End Sub
Sub SaveTheFile() 'এই সাবের কাজ হলো সংরক্ষণ ডায়লগ খুলে ফাইন সেভ করা।
    'সঠিকভাবে টেক্সট ফরমাটে সংরক্ষণের জন্য রিচ টেক্সট বাক্সের লেখা একটি লুকানো টেক্সট বাক্সে নিয়ে যেতে হবে।
    TextBox2.Text = RichTextBox1.Text
    Dim Save As New SaveFileDialog() 'ফাইল সংরক্ষণের ডায়লগ ক্লাসের নতুন ইন্সট্যান্ট।
    Dim myStreamWriter As System.IO.StreamWriter 'একটি সিস্টেম স্ট্রিমরিডার অবজেক্ট ডাইমেনশন করা।
    Save.Filter = "Text (*.txt)|*.txt|All files(*.*)|*.*" 'টেক্সট টাইপ ফিল্টার।
    Save.CheckPathExists = True 'চয়েজ করা ফাইল সংরক্ষণের প্যাথ সঠিক আছে কি না দেখতে হবে।
```

১৪তম পরিচ্ছেদ: ***Complete Application Tutorials***

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    Save.Title = "টেক্সট্ ফরমাটে ফাইল সংরক্ষণ করুন" 'সংরক্ষণ ডায়লগের শিরোনাম।
    Save.ShowDialog(Me) 'এখানে  মি  অর্থ স্বয়ং ডায়লগটি।
    Try
        'নিচের লাইনটি নিশ্চিত করে একই নামের ফাইল থেকে থাকলে তা প্রতিস্থাপন করা যাবে কি না? না গেলে ফাইল সংরক্ষণ হবে না।
        myStreamWriter = System.IO.File.AppendText(Save.FileName)
        myStreamWriter.Write(TextBox2.Text) 'এবার টেক্সট্ বাক্সে সংরক্ষিত পুরো টেক্সট্ ফাইলে লেখা হবে।
        myStreamWriter.Flush() 'মেমোরি থেকে ক্লাসটি মুছে দিতে হবে।
      Catch ex As Exception
    End Try
  End Sub
End Class
```

কোডের ব্যাখ্যঃ প্রথমেই লক্ষ করুন, বাংলা ইউনিকোডে প্রচুর কমেন্ট দিয়েছি। মাইক্রসফটকে ধন্যবাদ! তারা অপারেটিং সিস্টেমে বাংলায় উইনিকোড ক্যারাক্টার যুক্ত করে আমাদের বিরাট উপকার করেছেন। সুতরাং ভিজুয়েল বেসিক.নেট এর আইডিইতে আমরা অনায়াসেই বাংলা ইউনিকোডে লেখার সুযোগ পেয়েছি। প্রোগ্রামেও বাংলায় নির্দেশনা পর্যন্ত প্রদর্শন হওয়ার ব্যবস্থা হয়েছে। যদিও এখন পর্যন্ত রিচটেক্সটবক্স কন্ট্রোলের মধ্যে ইউনিকোড টাইপ করার সুযোগ তৈরি করে দেওয়া হয় নি। তবে আমরা আশাবাদী তা অচিরেই হয়ে যাবে।

যাক ফিরে আসি কোডের ব্যাখ্যায়। নিচে একে একে প্রতিটি সাবরুটিনের ব্যাখ্যা লাইন বাই লাইল তুলে ধরছি। প্রয়োজনে আলোচিত সাবরুটিন পুনরায় টাইপ করে দিয়েছি। লক্ষ করুন, সকল কোডই ফর্মের ক্লাস সাবরুটিনের ভেতর। আমরা কোনো আলাদা মডিউল বা ক্লাস তৈরি করি নি।

১. প্রথমেই একটি গ্লোবাল ইন্টেজার ভেরিয়েবল ডিক্লায়ার করা হয়েছে। এটি দ্বারাই আমরা প্রোগ্রাম চলাকালে রিচ টেক্সট্ বাক্সের সাইজ সঠিক রাখবো। কারণ ব্যবহারকারী প্রয়োজনে ফর্মের সাইজ রদবদল করতে পারেন।
২. `Form1_Load` সাবের কোডঃ এটির দু'টি কাজ আছেঃ (ক) রিচ টেক্সট্ বাক্সের সাইজ সেট করা ও (খ) চিচ টেক্সট্ বাক্সে সাউস রাইট-ক্লিককালে একটি কনটেক্সট্ মেনু প্রদর্শন করা। কনটেক্সট্ মেনুর মধ্যে কমি-পেস্ট-কাট- এ তিনটি সুবিধা থাকবে। ব্যবহারকারী লেখা বা লেখাংশ সিলেক্ট করে রাইট-ক্লিকের মাধ্যমে কনটেক্সট্ মেনু চয়েজ করে সহজেই এ তিনটি কাজের যে কোনো একটি করতে সক্ষম হবেন। সুতরাং সাবের প্রথম লাইন দ্বারা রিচ টেক্সট্ বাক্সের উচ্চতা ও প্রস্থ নির্ধারিত করে দেওয়া হয়েছে। দ্বিতীয় লাইনের কাজ হলো একে ফর্মের কেন্দ্রে নিয়ে আসা। আর তৃতীয় লাইন দ্বারা পরের সাবরুটিনটি কল করা হয়েছে যার কাজ হলো কনটেস্টট্ মেনু তৈরি করা।
৩. `InitializeMyContextMenu()` সাবের কোডঃ আগেই বলেছি, এটা আমাদের সৃষ্ট কনটেক্সট্ মেনু তৈরির সাবরুটিন। প্রথমে ডিফাইন করতে হবে একটি নতুন কনটেক্সট্ মেনু আইটেম অবজেক্ট। এরপর ডিফাইন করতে হবে তিনটি মেনু আইটেম নাম ও প্রতিটির সঙ্গে সম্পৃক্ত বোতামের একশন। তিন জোড়া কোড লাইনে এ কাজগুলো করা হয়েছে। সবশেষে প্রতিটি মেনু আইটেম দেখাতে হবে যখনই মাউসে রাইট-ক্লিক করা হবে এবং পুরো মেনুর সঙ্গে সমপৃক্ত করতে হবে রিচ টেক্সট্ বাক্সটি।
৪. `Create_Resize` নামক ফর্মের ইভেন্ট সাবের কোডঃ প্রথমত এই সাবরুটিন স্বয়ংক্রিভাবে তৈরি করতে হলে কোড লিখার ফিল্ডের উপরস্থ বায়ের কম্বো বাক্সে যেয়ে চয়েজ করুন ফর্ম [Form1 Events]। এবার ডানের কম্বো বাক্সে যেয়ে চয়েজ করুন রিসাইজ [Resize]। সাথে সাথেই এই সাবরুনিটি আসবে। এর ভেতরেই কোড লিখতে হবে। প্রথম ইফ-এন্ড ইফ [If - End If] ব্লকের কাজ হলো ফর্মের সাইজ পরিবর্তন হলে রিচ টেক্সবাক্সের সাইজও পরিবর্তন করা। চারটি লাইনের প্রথমটি দ্বারা ফর্মের প্রস্থের সাথে সামঞ্জস্য রেখে রিচ টেক্সট্ বাক্সোর প্রস্থও পরিবর্তন করা হয়েছে। উভয় দিকে ১০০ পন্টে বর্ডারও রাখা হয়েছে। শেষের লাইন দ্বারা নিশ্চিত করা হয়েছে ফর্মের মাঝখানে যেনো রিচ টেক্সট্ বাক্স থাকে। সাবের দ্বিতীয় ইফ-এন্ড ইফ [If - End If] অনুরূপ কাজ করে। তবে এখানে প্রস্থ বাড়ছে না বরং কমেছে- এটাই পার্থক্য। লক্ষ করুন, তার ক্লাসের ফেতর ফর্মকে ইঙ্গিত করতে হলে Me কিওয়ার্ড ব্যবহ্নর করা দরকার।
৪. `Create_SizeChanged` নামক ফর্মের ইভেন্ট সাবের কোডঃ এই কোডের কোড ও ক্রিয়া উপরে আলোচিত সাবের

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

মতোই। সুতরাং, শিক্ষার্থীতা একে ভালো করে দেখে নেবেন ও বাংলায় লেখা কমেন্ট পাঠ করবেন। আশাকরি বুঝতে অসুবিধা হবে না।

৫. UndoBt_Click ও RedoBt_Click নামক বোতামের ক্লিক একশন ইভেন্ট সাব: এই উভয় সাব একইভাবে কাজ করে। সাব দু'টি যথাক্রমে Undo() এবং Redo() নামক সিস্টেম ফাংশনের মাধ্যমে ক্রিয়া করে।

৬. BoldBt_Click ও ItalicBt_Click নামক বোতাম দু'টোর ক্লিক একশন ইভেন্ট সাব: এ দু'টোর কাজও অনুরূপ। সুতরাং কিছুটা ব্যাখ্যা দ্বারা উভয়টির ক্রিয়া বুঝা যাবে। উভয় কোডে ব্যবহৃত হয়েছে একটি If - ElseIf - End If ব্লক। উভয়টির মাঝে আবার আরেকটি Else - End If ব্লক আছে। দেখতেই পাচ্ছেন প্রথম ইফের কাজ হলো বল্ড থেকে থাকলে নরম্যাল করা- তবে দেখতে হবে সিলেকশন ইটালিকও আছে কি না- থাকলে ইটালিকসহ নরম্যাল করা চাই। অনুরূপ যদি বল্ড না থাকে তাহলে ইটালিক থাকলে সেটাসহ বল্ড বানাতে হবে। ElseIf অংশের কাজ এটি। আশারাখি শিক্ষার্থীরা লজিকটি বুঝে নিয়েছেন।

৭. FontBt_Click বোতামের ক্লিক একশন ইভেন্ট সাব: এটার কাজ ফন্ট পরিবর্তন করা। ভিজুয়্যাল বেসিক.নেট ফন্টলিস্টসহ একটি জানালা তৈরি করে দিয়েছে। সেখান থেকে সিস্টেমে ইন্সস্টল করা সকল ফন্টের মধ্যে যে কোনো একটি চয়েজ করা যায়। সিলেক্ট করা টেক্সট কিংবা ইনসারশন পয়েন্ট থেকে নতুন ফন্ট কাজ করবে। কোডের কাজ হলো ডায়লগটি খোলা ও সিলেক্টকরা ফন্ট পরিবর্তন করা।

৮. SizeUpBt_Click ও FontColBt_Click নামক বোতাম দু'টোর ক্লিক একশন ইভেন্ট সাব: এ দু'টো বোতামের সাব একইভাবে কাজ করে। প্রথমটি ৫ পয়েন্ট করে সিলেক্ট করা ফন্টের সাইজ বাড়াবে ও দ্বিতীয়টি ৫ পয়েন্ট করে কমাবে। এই যা। লক্ষ করুন সাইজ বাড়ানো ও কমানোর জন্য ব্যবহৃত হয়েছে দীর্ঘ এই একটিমাত্র কোড লাইন:

```
RichTextBox1.SelectionFont = New Font(RichTextBox1.SelectionFont.FontFamily, Int(RichTextBox1.SelectionFont.SizeInPoints - 5))
```

লাইনটির = সাইনের ডানের অংশ হলো একটি সিস্টেম ফাংশন। এর ভেতর প্যারামিটার হিসাবে আছে ব্যবহৃত ফন্ট ফ্যামিলি ও ইন্টেজার টাইপের সাইজ।

৯. FontColBt_Click নামক বোতামের ক্লিক একশন ইভেন্ট সাব: এ সাবটি উপরের (৭নং) ফন্ট পরিবর্তন সাবের মতোই কাজ করে। এ ক্ষেত্রে ডায়লগটি হলো কালার ডায়লগ। এই যা। পরের দু'টি বোতামের একশন ইভেন্ট কোডও কালার ডায়লগ খুলে যথাক্রমে সিলেক্ট করা লেখার হাইলাইট রং ও রিচ টেক্সট বাক্সের বেকগ্রাউন্ড রং ব্যবহারকারীর ইচ্ছেমাফিক পরিবর্ত করবে। সুতরাং এগুলো সম্পর্কে আর বেশি কিছু বলার নেই।

১০. AlinLBt, AlinCbt এবং AlinRbt নামক বোতাম তিনটির ক্লিক একশন ইভেন্ট সাব: এগুলোর কাজ কি তা-তো সকলেই বুঝতে পারছেন। তিনটি ক্ষেত্রেই কাঙ্ক্ষিত ফলাফল পেতে ব্যবহার করেছি HorizontalAlignment নামক সিস্টেম ফাংশন।

১১. CopyBt, CutBt এবং PstBt বোতাম তিনটির ক্লিক একশন ইভেন্ট সাব: এ তিনটি বোতামের কারণও আমাদের জানা। তবে লক্ষ করুন, প্রথম দু'টিতে Try - End Try ব্লকের ভেতর কোড লিখা হয়েছে। এর কারণ হলো ক্লিপ বোর্ডে সবকিছু যেতে না-ও পারে। কপি এবং কাট- উভয় বোতামের কাজ হলো ক্লিপবোর্ডে সিলেকটেড টেক্সট যথাক্রমে কপি বা মুছে কেটে নেওয়া। সিলেকশন যদি না থাকে- কিংবা টেক্সট ছাড়া অন্য কিছু থাকে তাহলে যে কোডিং সমস্যা হবে সেটা এড়াতে উক্ত Try - End Try ব্লক ব্যবহৃত হয়েছে। কপি করতে যেয়ে আমরা সিস্টেমের Clipboard নামক ফাংশনটি ব্যবহার করেছি। এর প্যারামিটার হলো সিলেকটেড টেক্সট। লক্ষ করুন দ্বিতীয় বোতামের কোডে অতিরিক্ত একটি লাইন দ্বারা সিলেকটেড টেক্সট মুছে দেওয়ার নির্দেশনা দিয়েছি। তৃতীয় বোতামের কোড কিছুটা ভিন্ন। সেখানে একটি If - End If ব্লকের ভেতর সিস্টেমের Paste কিওয়াডটি ব্যবহার করেছি। ক্লিপবোর্ডে কিছু আছে কি না তা দেখেই পেস্ট করতে হবে।

১২. OpenFileBt ও SaveFileBt বোতামদ্বয়ের ক্লিক একশন ইভেন্ট সাব: এ দু'টোর একশন প্রায় একই। প্রথমটি যে ডায়লগ খুলে তাহলো OpenFileDialog() ডায়লগ। আর দ্বিতীয়টি খুলে SaveFileDialog() ডায়লগটি। উভয়টিই সিস্টেমের ফাংশন। লক্ষ করুন, ডায়লগ খোলার জন্য নতুন ডায়লগ ডিক্লারেশন পরে আমরা টাইপ এক্সটেনশন ও ফিল্টার নির্দিষ্ট করেছি। আমাদের উদ্দেশ্য শুধুমাত্র .rtf ফাইল নিয়ে কাজ করা। এটাই হচ্ছে রিচ টেক্সট বাক্সের ফাইল টাইপ। প্রথম সাােব আমরা ইফ - এন্ড ইফ টেস্টের মাধ্যমে নিশ্চিত করেছি যে ব্যবহারকারী ফাইল খুলেছেন ও ফাইলটি শূন্য নয়। এরপর একটি মাত্র লাইনে নির্দেশ দেওয়া হয়েছে ফাইল খুলে সব লেখা রিচ টেক্সট বাক্সে প্রদর্শন করতে। এর

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

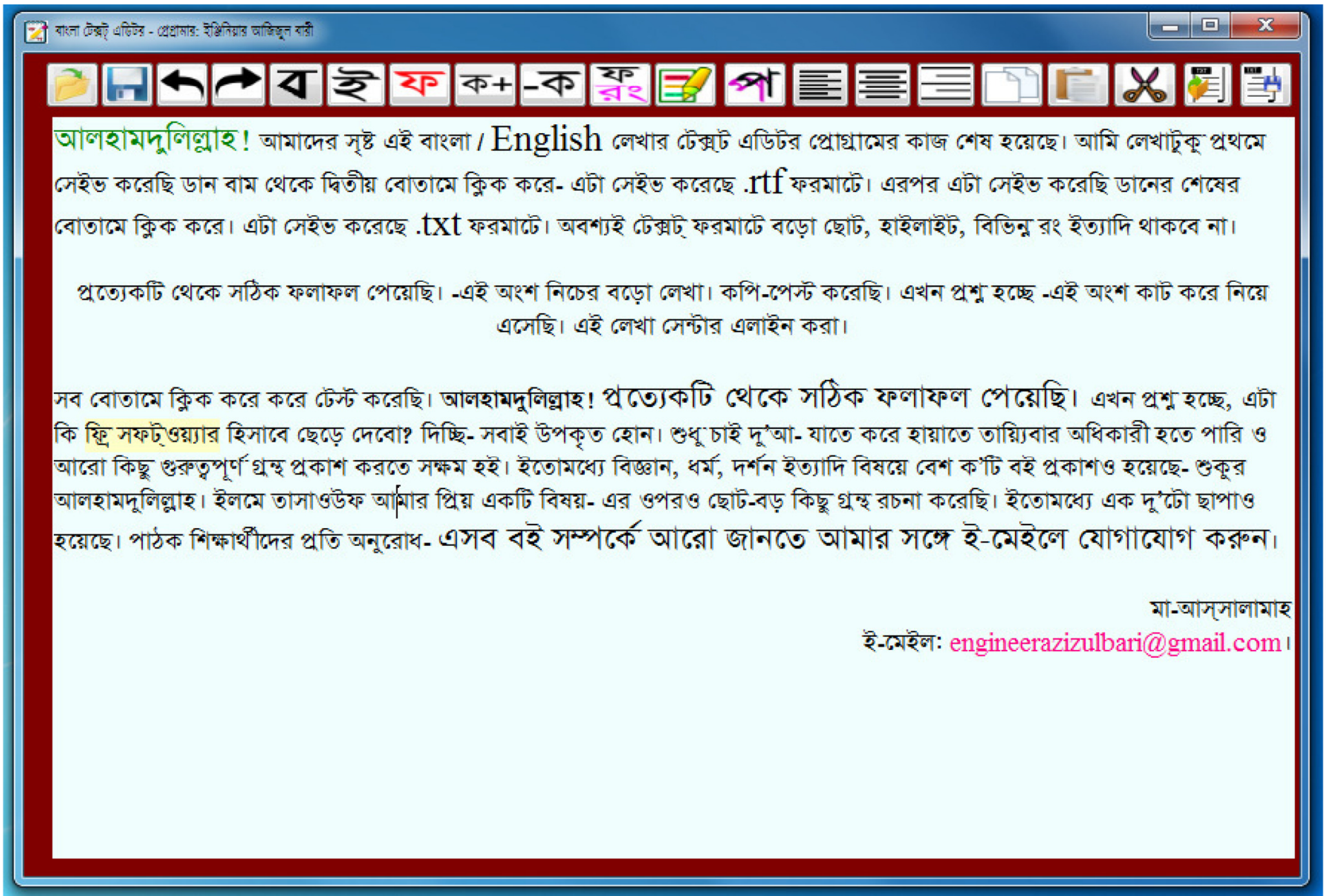
পরের লাইন ফাইলের প্যাথ নাম জানালার আইকনের পাশে প্রদর্শন করবে।
দ্বিতীয় সাবের কাজও অনুরূপ। তবে লক্ষ করুন, সেইভ করা ফাইল কোথায় সংরক্ষিত হয়েছে সে প্যাথ নাম জানালার আইকনের পাশে লিখার নির্দেন দেই নি। তবে চাইলে দেওয়া যেতো।
১৩. Form1_Closing ইভেন্ট সাবের কোড: কম্পিউটারের প্রতিটি কাজ সম্পাদনে একটি ঘটনা বা ইভেন্ট সিস্টেম সফটওয়্যার থেকে আসে। সফটওয়্যার ডেভেলাপার হিসেবে আমাদের কাজ হলো এসব ইভেন্টকে ধরে ইচ্ছেমতো কাজে লাগানো। ইভেন্ট জেনারেট হয় কিবোর্ড, মাউস ইত্যাদি থেকে। ফর্ম বন্ধ করার সময় আমরা মাউস দ্বারা একটি আইকনে [জানালার ডান পাশের x-ক্রসে] ক্লিক করি। এতে একটি সিস্টেম ইভেন্ট সক্রিয় হয়। এটির নামই Form1_Closing ইভেন্ট। আমরা একটি সাবে একে ধরে কিছু নির্দেশনা দিয়েছি। একটি ডবল নেস্টেড্ ইফ - এন্ড ইফ ব্লকে এই কোড লিখেছি। আমাদের উদ্দেশ্য হলো, রিচ টেক্সট বাক্সে কিছু লেখা থাকলে তা সংরক্ষণ করা হবে কি না সেটা ব্যবহারকারীকে প্রশ্ন করা। জবাব 'হ্যাঁ' হলে সংরক্ষণ করে প্রোগ্রাম ক্লোজ করা- অন্যথায় এমনিতেই ক্লোজ করা। লক্ষ করুন, সেইভ করার জন্য আমরা পুনরায় সেইভ ডায়লগ খোলার কোড এখানে লিখি নি- বরং সেইভ বোতামে স্বয়ংক্রিয়ভাবে ক্লিক ইভেন্ট জেনারেট করার নির্দেশ দিয়েছি। সাবের কোড রান করা শেষ হয়ে গেলেই কিন্তু প্রোগ্রাম বন্ধ হয়ে যাবে।
১৪. আমাদের ২০টি বোতামের ওপর ব্যবহারকারী মাউস নিয়ে গেলে বাংলায় কিছু তথ্যমূলক লেখা দেখতে পাবেন। অর্থাৎ কোন্ বোতামের কাজ কোনটি তা জানতে পারবেন। যে উপায়ে এটা জানানো হয় তাকে বলে SetToolTip। এটা সিস্টেমের একটি ফাংশন যার প্যারামিটার হলো অবজেক্ট বা বস্তুর নাম [এ ক্ষেত্রে বোতামের] ও পপাপ বাক্সে কি লিখা থাকবে সে স্ট্রিং। অর্থাৎ যদি এভাবে ফাংশন ব্যবহার করি: `SetToolTip (Button1, "□ □□□□□□□□□□□ □□□□□")` তাহলে ঐ বোতামের ওপর মাউস নিলেই ছোট্ট একটি পপাপ জানালা আসবে যেখানে লিখা থাকবে, □ □□□□□□□□□□□ □□ □□□□ মোটকথা দ্বিতীয় প্যারামিটারের ভেতর যা-ই লিখা থাকুক না কেনো, তা জানালায় প্রদর্শিত হবে। আমরা ২০টি বোতামের টুল টিপ লিখে দিয়েছি যাতেকরে ব্যবহারকারী জানতে পারেন কোন্ বোতামে ক্লিক করলে কোন্ একশন হবে।

১৫. `OpenTxtBt`, `SaveTxtBt` ও `SaveTheFile()` নামক ফাংশন সাবের ব্যাখ্যা: প্রথম দু'টি হলো বোতামের ক্লিক ইভেন্ট সাব ও শেষেরটি আমাদের সৃষ্ট একটি ফাংশন। আমরা চাই ব্যবহারকারী প্রয়োজনে লেখাটুকু টেক্সট ফরমাটে সেইভ করা ও একই ফরমাটের লেখা রিচ টেক্সট বাক্সে নিয়ে আসার সুযোগ পাবেন। এই তিনটি সাবের কাজ তা-ই। প্রথমটি ও শেষেরটি মূলত আগে ব্যাখ্যাত OpenFileBt ও SaveFileBt বোতামদ্বয়ের সাবের অনুরূপ। এখানে পার্থক্য হলো টাইপ ও ফিল্টার .rtf এর বদলে .txt লিখেছি। তবে লক্ষ করুন, সংরক্ষণ নির্দেশের পূর্বে পুরো লেখাটুকু একটি টেক্সট্ ফরমাটের টেক্সট্ বাক্সে নিয়ে আসা হয়েছে। এই বাক্সটি কিন্তু লুকানো। এতে নিয়ে আসার ফলে টেক্সট্ ফরমেটিং সঠিক হবে। যাক, কোডের ব্যাখ্যা এটুকুই- যেহেতু কোডের মধ্যেই বাংলায় ব্যাখ্যামূলক কমেন্ট করেছি তাই আর অতিরিক্ত বলার প্রয়োজন নেই। আমি আশারাখি শিক্ষার্থীরা কোড বুঝতে খুব একটা সমস্যায় পড়বেন না। এবার দেখে নিই প্রোগ্রমাটি একবার রান করে- কেমন? পরের পৃষ্ঠার চিত্রটি দেখুন। ভেতরে কি লেখা আছে তা-ও দয়া করে পাঠ করবেন!

১৬তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

আমরা মোট ৪টি ব্যবহারযোগ্য মোটামুটি জটিল এ্যাপলিকেশন তৈরি করতে সক্ষম হয়েছি। আরো দু'টি প্রোগ্রাম ডেভেলাপ করবো শিক্ষার্থীদের সুবিধার জন্য। তবে এক্ষুণি নয়। পরের পৃষ্ঠাসমূহে ভিজুয়্যাল বেসিক.নেট -এর একটি শক্তিশালী ডেভেলাপমেন্ট টুল সম্পর্কে কিছুটা আলোকপাত করার প্রয়াস পাচ্ছি। চাক্ষুষ্মান শিক্ষার্থীদের মনে হয়তো প্রশ্ন জেগেছে, অনেক গুরুত্বপূর্ণ কম্পিউটার সফ্‌টওয়্যার সিস্টেম 'ডটাবেইজ' সম্পর্কে এই নীরবতা কেনো? এ



প্রশ্নের জবাবেই পরের পুরো অনুচ্ছেদটির অবতারণা। ডাটাবেইজ করার জন্য পর্যাপ্ত টুলস্ ও ফাংশন অবশ্যই .নেট-এ আছে। আমরা অবশ্যই এডভান্স ডাটাবেইজ প্রোগ্রামিং সম্পর্কে কিছু বলবো না। এটা এই গ্রন্থের স্কোপের বাইরে। তবে বেসিক বিষয়গুলো অবশ্যই জেনে নেবেন। আমরা শেষের দিকে একটি বাংলা ঠিকানাবই ডাটাবেইস তৈরিও করবো। শিক্ষার্থীরা মৌলিক বিষয়গুলো রপ্ত করে নিতে পারলে, ধীরে ধীরে এডভান্স ডাটাবেইজও ডেভেলাপ করতে পারবেন নিশ্চয়ই। কারণ এটকু পরই দেখতে পাবেন, মাইক্রসফটের 'Access' ডাটাবেইজের টেবিল কিভাবে অতি

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

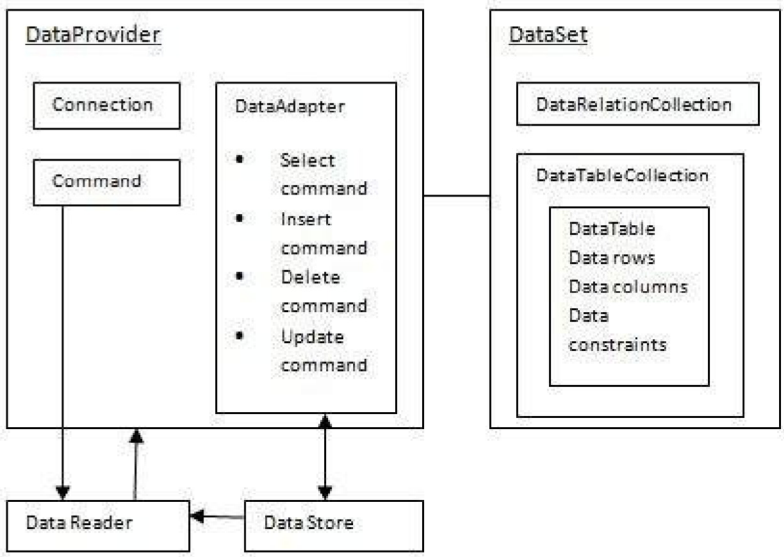
সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

অনেকেই ভাবেন, ডাটাবেইজ প্রোগ্রামিং বুঝি খব জটিল-কঠিন। বাস্তবে তা মোটেই নয়। অন্তত, ভিজুয়্যাল বেসিক.নেট রিলিজ হওয়ার পর থেকে এই ধারণা যে পাল্টে গেছে তা নিশ্চিত করে বলা যায়। এ পরিচ্ছেদ থেকে বিষয়টি অবশ্যই শিক্ষার্থীদের নিকট স্পষ্ট হবে। বাস্তবে সফটওয়্যার ডেভেলাপমেন্ট মূলত একটি চিত্তাকর্ষক ক্রিয়ার নাম। তবে যা প্রয়োজন তাহলো, শিক্ষাগ্রহ ও কিছুটা সময়। যাক, এবার আমরা একে একে ডাটাবেইজ প্রোগ্রামিংয়ের ওপর উদাহরণসহ বিস্তারিত আলোচনা করতে চাচ্ছি।

১. এক্টিভএক্স ডাটা অবজেক্ট.নেট [সংক্ষেপে: ADO.Net]

এ্যাপলিকেশন দু'টি উপায়ে সংরক্ষিত ডাটা বা তথ্য নিয়ে কাজ করতে পারে: ১. ব্যবহার-বান্ধব উপায়ে প্রদর্শনের জন্য সংরক্ষিত ডাটা নিয়ে আসা। ২. সংরক্ষিত ডাটাকে প্রয়োজনে পরিবর্তন, পরিবর্ধন, সংযোজন ও মুছে দেওয়া এবং পুনরায় সংরক্ষণ নিশ্চিত করা। মাইক্রসফট .নেট সৃষ্টিতে এডো.নেট [ADO.Net] নামক একটি মডেল তৈরি করেছে। এটা মূলত .নেট এর ফ্রেমওয়ার্কের একটি অংশ। ফলে .নেট কোনো ডাটাসূত্র যেমন মাইক্রসফটের Access থেকে ডাটা সহজেই নিয়ে এসে কাজ করতে পারে।

(ক) ADO.Net Object Model - এডো.নেট অবজেক্ট মডেল: এটা মূলত কাঠামোগত একটি প্রসেসের নাম। ডাটাবেইজের সঙ্গে সম্পৃক্ত বিভিন্ন উপাদানে কিভাবে তথ্য চলাফেরা করে তারই একটি উপায়-অবলম্বন। আমরা একে চিত্রের মাধ্যমে তুলে ধরতে পারি। যেমন:



লক্ষ করুন, ডাটা কোনো ডাটাবেইজ বা স্টোরে সংরক্ষিত থাকে। এটা DataProvider দ্বারা বের করে নিয়ে আসা হয়। এই প্রভাইডারের ভেতর বেশ ক'টি অংশ আছে যেগুলোর কাজ হলো ডাটা এ্যাপলিকেশনে নিয়ে আসা যাতেকরে আপডেট করা যায়।

এ্যাপলিকেশনে ডাটা গ্রহণ করে হয় DataSet না হয় DataReader এর মাধ্যমে।

* DataSet তথ্য সংরক্ষণ করে রাখে একটি আলাদা ক্যাশিতে। সেটা এ্যাপলিকেশন নিয়ে আসে।

* DataReader এ্যাপলিকেশনে ডাটা সংযোজন করে শুধুমাত্র পাঠ [read-only] ও অগ্রগামী [forward-only] তথ্য হিসাবে।

(খ) DataProvider - তথ্য সরবরাহকারী : এটা ব্যবহৃত হয় ডাটাবেইজের সঙ্গে সংযুক্তি, কমান্ড এক্সেকিউট, তথ্য আদান-প্রদান, ডাটাসেটে সংরক্ষণ, পড়া ও পরিবর্তন ইত্যাদি কাজের জন্য। ADO.Net -এ ডাটা প্রভাইডারে আছে চারটি অবজেক্ট। যথা:-

(১) Connection - সংযুক্তি: বুঝাই যাচ্ছে এ অংশের কাজ হলো কোনো ডাটা সূত্রের সঙ্গে সংযুক্তি সৃষ্টি করা।

(২) Command - নির্দেশ: কমান্ড হলো কোনো SQL [Structured Query Language] স্টেটমেন্ট কিংবা স্টোরকরা প্রসিডোর। এরই মাধ্যমে ডাটা সূত্রের তথ্যাদি পরিবর্তন, মুছা, সংরক্ষণ ও সংযোজন ইত্যাদি কাজ করা হয়।

(৩) DataReader - তথ্য পাঠক: বুঝাই যাচ্ছে এ অংশের কাজ হলো কোনো ডাটা সূত্র থেকে শুধুমাত্র পাঠ ও

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

অগ্রগামী মোডে কাজ করা। তথ্যাদি পরিবর্তন করা যাবে না।

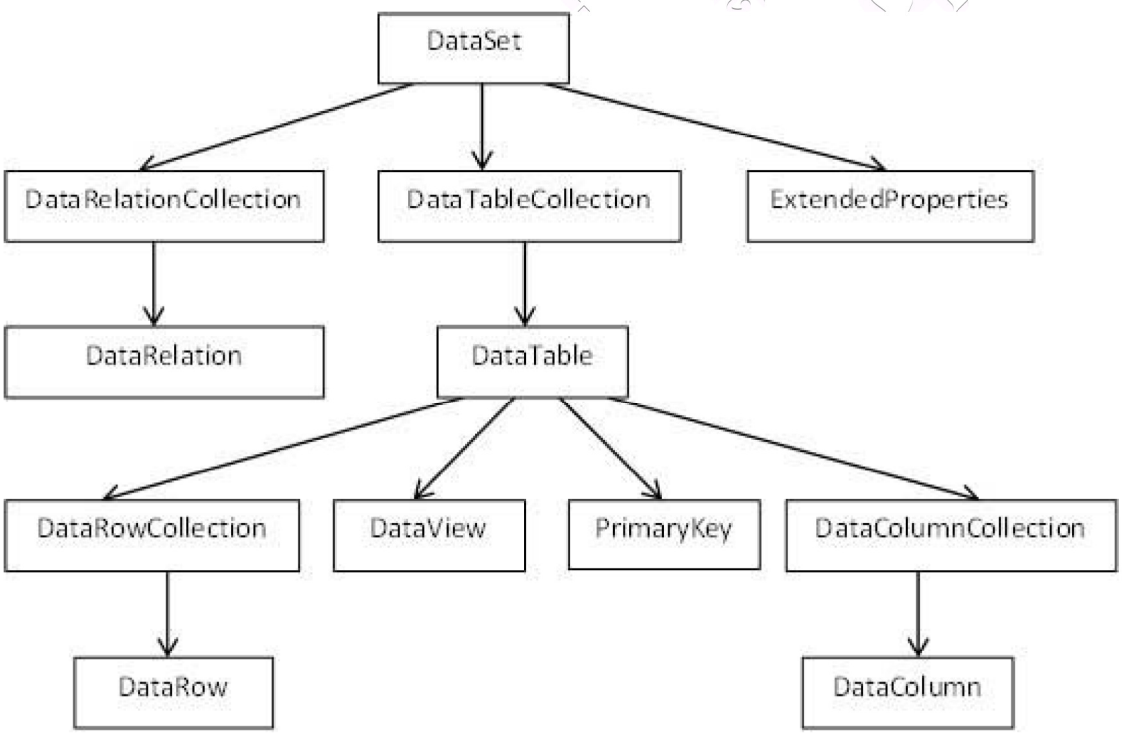
(৪) DataAdapter - তথ্য অভিযোজক: এটা ADO.Net এর একটি গুরুত্বপূর্ণ বস্তু। এই তথ্য অভিযোজকের মাধ্যমেই যাতবীয় ডাটা ডাটাবেইজ থেকে আদান-প্রদান হয়। এটা ডাটা ডাটাবেইজ থেকে ডাটাসেটে নিয়ে আসে এবং এটাই আবার ডাটা আপডেট করে। ডাটাসেটে রদবদল হলে এই ডাটাএডাপটারই ডাটাবেইজে পরিবর্তন আনয়ন ও সংরক্ষণ করে।

ADO.Net -এ বেশ ক'টি ডাটা সরবরাহকারী আছে। যেমন:

* SQL সার্ভারের জন্য .Net Framework ডাটা সরবরাহকারী।
* OLE [Object Linking and Embedding] ডাটাবেইজের জন্য .Net Framework ডাটা সরবরাহকারী।
* ODBC [Open Database Connectivity] ডাটাবেইজের জন্য .Net Framework ডাটা সরবরাহকারী।
* Oracle [Oak Ridge Automatic *Computer* and Logical Engine] ডাটাবেইজের জন্য .Net Framework ডাটা সরবরাহকারী।
* EntityClient সরবরাহকারী- এর মাধ্যমে EDM [Entity Data Model] এ্যাপলিকেশন ব্যাবহার করে ডাটা সংগ্রহের ব্যবস্থা।

DataSet

এটা হলো মেমোরির -মধ্যে তথ্য উপস্থাপনের একটি উপায়। এটা অসংযুক্ত, ক্যাশিতে সংরক্ষিত রেকর্ড সেট যা কোনো ডাটাবেইজ থেকে প্রাপ্ত হওয়া যায়। যখনই কোনো ডাটাবেইজের সঙ্গে সংযুক্তি সৃষ্টি হয় তখন, ডাটা এডাপটারে একটি ডাটাসেট তৈরি করে এতে সংরক্ষণ করে। অপরদিকে ডাটাবেইজের সঙ্গে সংযুক্তি বন্ধ হয়ে যায়। এটাকে বলে 'disconnected architecture'। ডাটাসেট একটি Virtual ডাটাবেইজ হিসেবে কাজ করে। এতে থাকে টেবিল, রো এবং কলাম। নিচের চিত্র থেকে হয়তো এই ডাটাসেট অবজেক্ট মডেলটি সম্পর্কে আরো বেশি অবগতি সম্ভব হবে।



১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

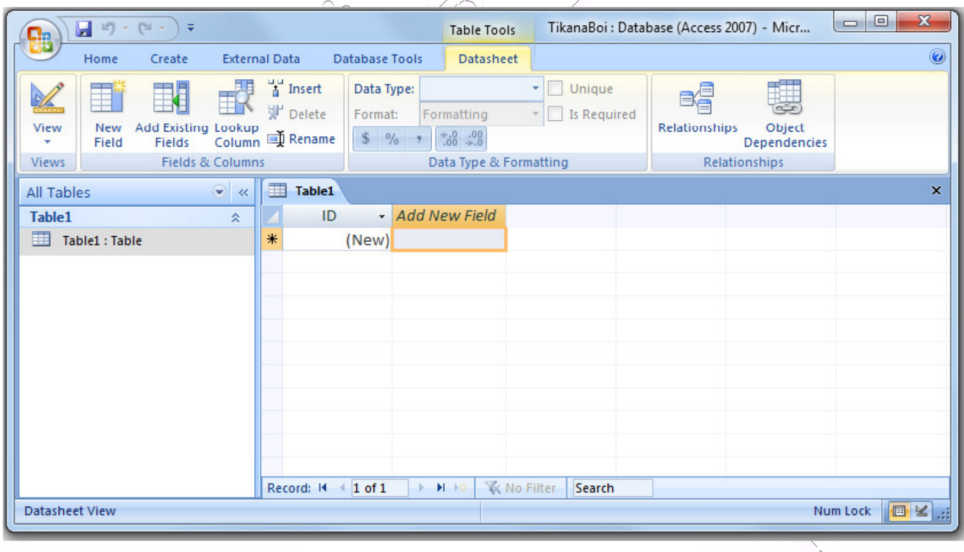
যাক, খুব বেশি টেকনিক্যাল কথা নিয়ে মাথা ঘামাবার প্রয়োজন নেই- উদাহরণের মাধ্যমে যে কোনো বিষয় সম্পর্কে জ্ঞানার্জন একটি উত্তম উপায়। এছাড়া ডাটাবেইজের মতো একটি উচ্চ পর্যায়ের কম্পিউটার প্রোগ্রামিং সম্পর্কে এ গ্রন্থে খুব একটা বেশি বলাও সম্ভব নয়। এ বিষয়ের উপর আলাদাভাবে অনেক গ্রন্থাদি রচিত হয়েছে। আমাদের উদ্দেশ্য হলো, ভিজুয়্যাল বেসিক.নেট ও ডাটাবেইজ এ দু'য়ের মধ্যে তথ্য আদান-প্রদান সম্পর্কে মৌলিক কিছু জ্ঞানার্জন- এই যা। সুতরাং আসুন, প্রথমে একটি উদাহরণ প্রজেক্ট তৈরি করি।

১. ভিজুয়্যাল বেসিক.নেট ও ডাটাবেইজ: একটি ঠিকানা বই ডাটাবেইজ প্রোগ্রাম

ঠিকানা বই ডাটাবেইজ প্রোগ্রামের জন্য মাইক্রসফটের Access ২০০৭ কিংবা এর পরের কোনো ভার্সন আপনার কম্পিউটারে থাকতে হবে। নিম্নের পদক্ষেপগুলো অনুসরণ করে ডাটাবেইজ প্রোগ্রামটি তৈরি করুন।

১. প্রোগ্রাম ফাইলে যেয়ে Access খুলুন। একটি নতুন ডাটাবেইজ তৈরি করুন। এর নামকরণ করুন 'TikanaBoi'। যেখানে ইচ্ছা এটা সেইভ করুন। তবে মনে রাখবেন, পরবর্তিতে এই ফাইলের প্যাথনেইম আমাদের দরকার।

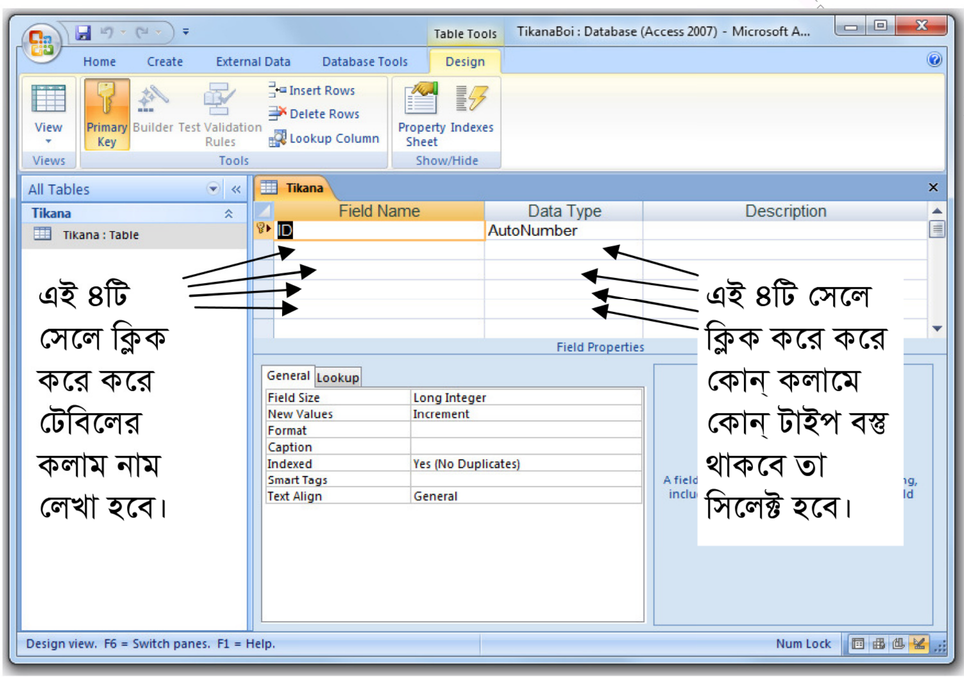
২. প্রথমেই আপনি নিচের চিত্র মুতাবিক জানালাটা দেখতে পাবেন।



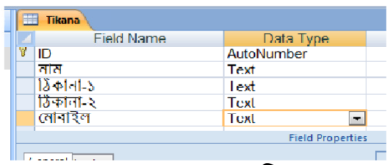
৩. বায়ের উপরের কোণে ডানে চিত্রিত আইকনে ক্লিক করুন। যে জানালা আসবে তাতে টেবিলের নাম দিন: 'Tikana'।



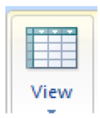
৪. এবার যে জানালাটি আসবে তার একটি চিত্র নিচে দেখানো হয়েছে।



৫. কোন্ কোন্ সেলে আমরা লিখবো তা চিত্রে দেখানো হয়েছে এ্যারো দ্বারা। Field Name কলামের দ্বিতীয় আইটেম হবে: 'নাম', পরেরটি হবে 'ঠিকানা-১', এরপর হবে 'ঠিকানা-২' এবং সবশেষের সেলে হবে 'মোবাইল'। Data Type কলামের চারটি সেলেই 'Text' শব্দটি থাকবে। দেখুন নিচের চিত্রটি। ব্যস্ ডিজাইন জানালার কাজ



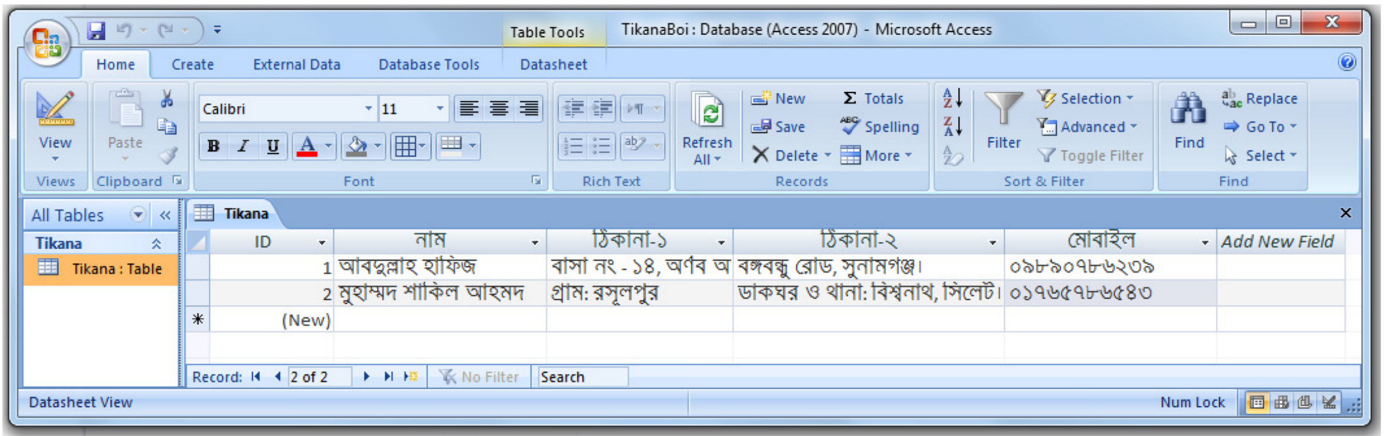
শেষ হলো। এবার ক্লিক করুন উপরের কোণায় স্থাপিত নিচে চিত্রিত আইকনে। একটি ডায়লগ আসবে সেইভ করার জন্য। এরপর আসবে টেবিল ভিউ, যেখানো আমরা তথ্য লিখতে পারি।



১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

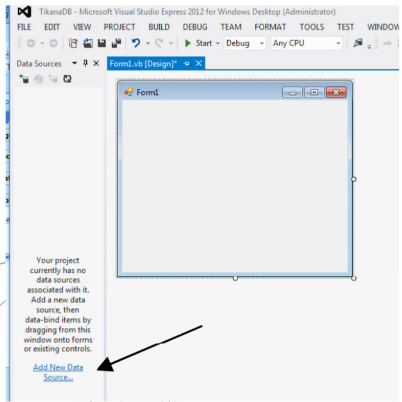
সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

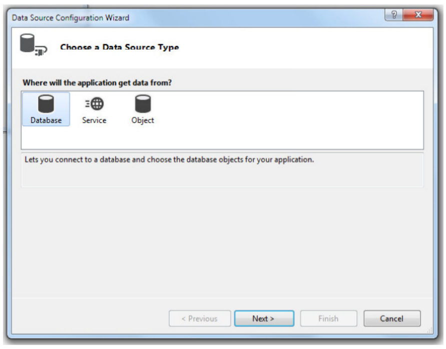
৬. প্রতিটি কলামের প্রতিটি সেলে বাংলা ইউনিকোডে লিখুন। কি লিখবেন ও কোথায় লিখবেন তা নিচের চিত্রে দেওয়া হয়েছে। অবশ্য আপনি চাইলে নাম, ঠিকানা ইত্যাদি ভিন্নও লিখতে পারেন। সেইভ করে বন্ধ করুন।



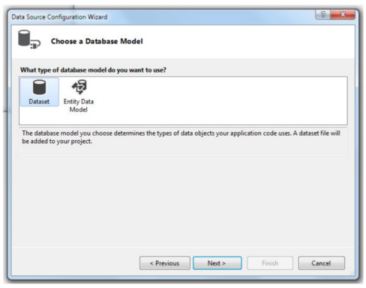
৭. আমাদের ডাটাবেইজ হয়ে গেছে। এবার আমরা ভিজুয়েল এক্সপ্রেস স্টুডিও থেকে একটি নতুন উইন্ডোজ সলুশন তৈরি করবো। এর নামকরণ করুন: 'TikanaDB'। সেইভ করুন।

৮. ফর্মের বায়ের দিকে থাকবে 'Data Sources' নামক একটি ট্যাব। [না থাকলে ক্লিক করুন: **View > Other Windows > Data Sources**]। ক্লিক করুন ডানের চিত্রে দেখানো লিংকে। এবার বায়ের চিত্রে দেখানো জানালাটি আসবে।

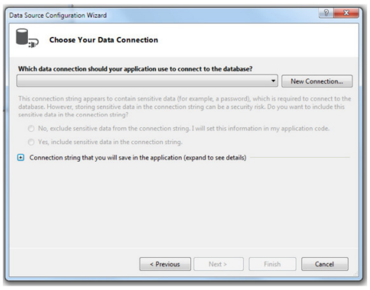




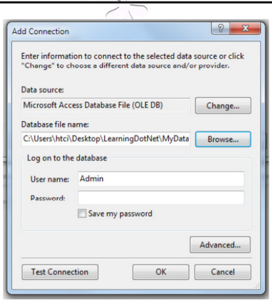
৯. ডাটাবেইজ সিলেক্ট করা থাকবে। Next বোতামে ক্লিক করুন। এর পরে যেসব জানালা আছে সেগুলো একে একে কিভাবে চয়েজ করবেন তা নিচের ক'টি চিত্রে দেখানো হয়েছে। হাইলাইট করা আইটেমগুলো চয়েজ করে করে সর্বশেষ জানালায় যেয়ে 'Finish' বোতামে ক্লিক করুন। ব্যস 'TikanaBoi' -এর সঙ্গে কানেকশন হয়ে যাবে।



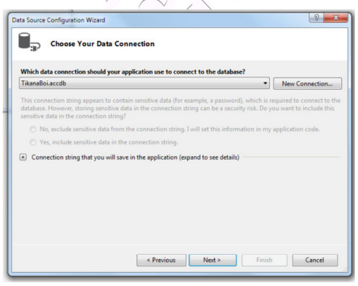
ক্লিক করুন: Next



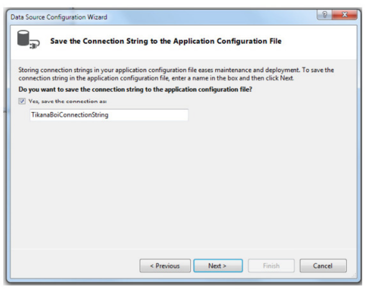
ক্লিক করুন: New Connection...



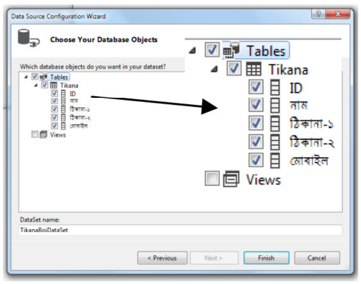
ক্লিক করুন: Browse...



ক্লিক করুন: Next



ক্লিক করুন: Next



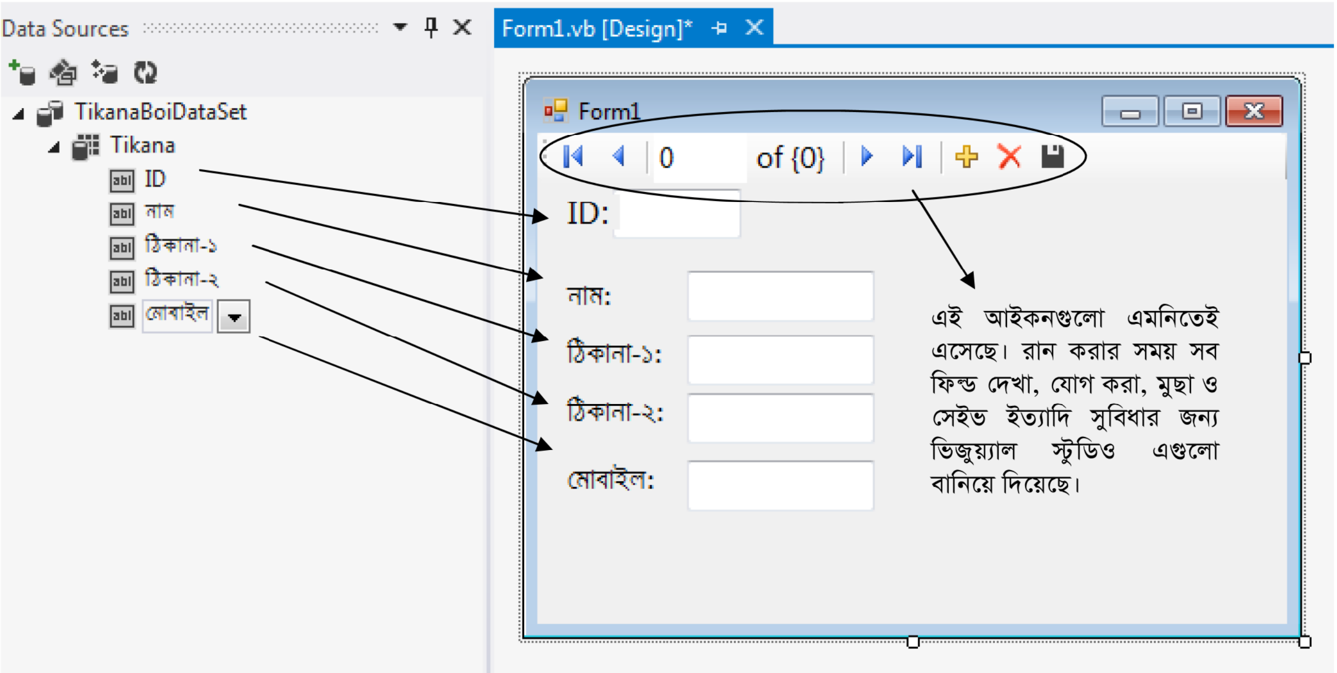
ক্লিক করুন: Finish

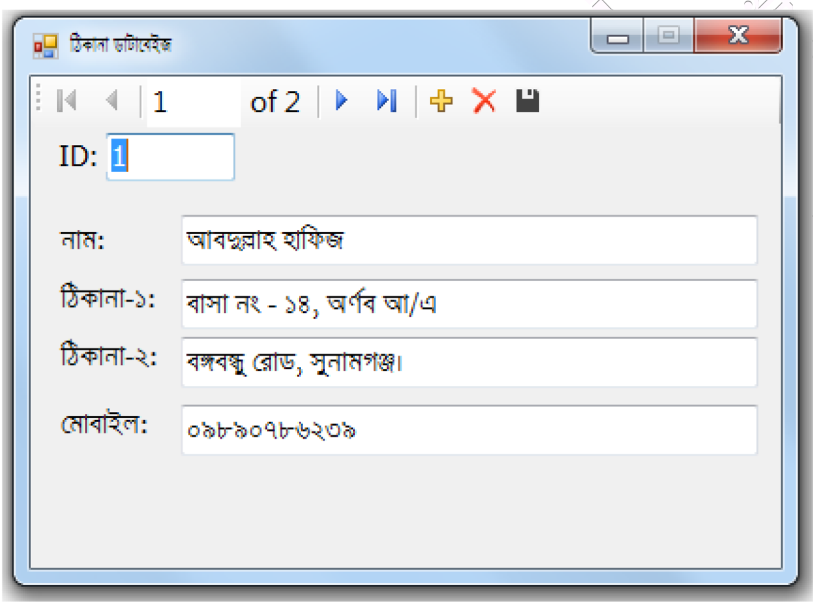
লক্ষ করুন, বায়ের চিত্রে টেবিল ও এর সবগুলো কলাম সিলেক্ট করা আছে। সুতরাং **'Finish'** বোতামে ক্লিকের পূর্বে এগুলো সিলেক্ট করতে ভুলবেন না। আমরা ইনসেটে সিলেকশনকে বড়ো করে দেখিয়েছি। উইজার্ড বন্ধ হওয়ার পর বায়ের ট্যাবের 'Data Sources'-এ ডাটাবেইজের তথ্য আসবে। ***নোট: নিউ কানেকশন [৩নং] ডায়লগ বাদে একটি ডায়লগ আসলে 'No' ক্লিক করবেন।***

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

১০. সফলভাবে কানেকশন হওয়ার পর আমরা ফর্মের মধ্যে টেবিলের বিভিন্ন বস্তু টেনে টেনে নিয়ে এসে যেখানে ইচ্ছে বসাতে পারি। নিচের চিত্র দেখুন। আপনি প্রতিটি আইটেম সিলেক্ট করে লেফ্‌ট-ক্লিক করে টেনে ফর্মের উপর এনে ছেড়ে দেবেন।





লক্ষ করুন, এ পর্যন্ত যাকিছু করলাম, তাতে এক লাইনের কোডিং করতে হয় নি। বাস্তবে ভিজুয়্যাল স্টুডিও আমাদের জন্য কোডিং করে দিয়েছে। কোডিং নিজে না করলেও যাকিছু উইজার্ডের মাধ্যমে হয়েছে- তা-ই আমাদের ডাটাবেইজ দেখার জন্য যথেষ্ট। এ কথার সত্যতা যাচাই করতে হলে এ পর্যায়ে প্রোগ্রামটি একবার রান করুন। বায়ের চিত্রটি দেখুন। আপনি চাইলে + আইকনে ক্লিক করে নতুন রেকর্ড তৈরি করুন। চারটি ফিল্ড পূরণ করে ডানের সেইভ আইকনে ক্লিক দিন। এরপর প্রোগ্রাম বন্ধ করে আবার খুলুন। দেখবেন, তিনটি রেকর্ড দেখাচ্ছে। শেষের [৩নং] রেকর্ডটি দেখুন।

যাক, কোডিং ছাড়াই একটি শক্তিশালী ডাটাবেইজ তৈরি করতে আমরা সক্ষম হয়েছি। তবে আমাদের ইচ্ছে তো, কোডিং শেখা। সুতরাং আসুন, কিভাবে কোডিংয়ের মাধ্যমে অনুরূপ ডাটাবেইজ তৈরি করা যায় তা একবার তলিয়ে দেখি।

১৫তম পরিচ্ছেদ: ***Database Programming Made Simple***

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

ডাটাবেইজ প্রোগ্রাম কোডিং

আগের ক'টি পৃষ্ঠাসমূহে ভিজুয়্যাল স্টুডিও কর্তৃক সৃষ্ট উইজার্ড দ্বারা কিভাবে একটি শক্তিশালী ডাটাবেইজ এ্যাপলিকেশন তৈরি করা যায় তা আমরা দেখেছি। এবার দেখবো ঠিক কিভাবে এ কাজগুলো কোর্ডের মাধ্যমে করা যায়। এ প্রজেক্টের জন্যও আমার Access -এ তৈরি আমাদের 'TikanaBoiDB' ডাটাবেইজ ও এর টেবিলই ব্যবহার করবো। এর একটি কপি আমরা ডেক্সটপে রেখে দেবো। ইউন্ডোজ ফর্ম প্রজেক্ট তৈরিতে নিচের পদক্ষেপগুলো অনুসরণ করুন।

(ক) কোনো প্রজেক্ট খোলা থাকলে বন্ধ করুন। নতুন প্রজেক্ট সৃষ্টি করুন। এর নামকরণ করুন 'TikanaBoiDB2'।

(খ) কানেকশন অবজেক্ট [Connection Object]: যে কোনো ডাটাবেইজের সঙ্গে সংযুক্তির জন্য প্রয়োজন এ বস্তুটি। সুতরাং এ সম্পর্কে আরো জানা জরুরী। এ পরিচ্ছেদের শুরুতেই বলেছি কানেকশন অবজেক্ট কয়েক ধরনের আছে। কোন্ ধরনের কানেকশন অবজেক্ট ব্যবহৃত হবে তা অবশ্যই নির্ভর করে কোন্ ধরনের ডাটাবেইজ নিয়ে আমরা কাজ করবো। আর যেহেতু আমরা Access ডাটাবেইজ নিয়ে কাজ করতে চাই তাই আমাদেরকে ব্যবহার করতে হবে OLE connection object। আগেই বলেছি OLE শব্দের অর্থ হলো, **O**bject **L**inking and **E**mbedding। মূলত এ'টি হচ্ছে অনেকগুলো COM [**C**omponant **O**bject **M**odel] বস্তুর সমষ্টি। সাধারণত অএলই-এর মাধ্যমে ডাটাবেইজের সঙ্গে সংযুক্তি স্থাপিত হয়। তবে এটার মাধ্যমে আরো অনেক সংযুক্তি হয়ে থাকে- যেমন, টেক্সট ফাইল, এসকিউএল সার্ভার, ই-মেইল ইত্যাদি। বেশ ক'টি ভিন্ন OLE অবজেক্ট আছে- এগুলোকে বলে প্রভাইডার [Provider]। আমরা যেটি ব্যবহার করবো তার নাম '**ACE**'।

(গ) যাক, এবার আপনার ফর্মে একটি বোতাম টুলবাক্স থেকে যুক্ত করুন। এর নামকরণ করুন [ইউনিকোডে] ' □□□ □ □□'।

(ঘ) বোতামে ডবল ক্লিক করে এর ক্লিক ইভেন্ট সাবটি খুলুন। এতে যুক্ত করুন নিচের কোড লাইনটি:

```
Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
```

(ঙ) কানেকশন অবজেক্টের সঙ্গে সম্পৃক্ত আছে বিভিন্ন প্রোপার্টি। এর মধ্যে একটি হচ্ছে ConnectionString প্রেপার্টি। এটা অনেক প্যারামিটার নিতে পারে। আমাদের ক্ষেত্রে অবশ্য কয়েকটি মাত্র দরকার। যেমন: প্রথমত একে বলতে হবে ডাটাবেইজ কানেকশন হবে ও দ্বিতীয়ত কোথায় সে ডাটাবেইজ আছে তা অবগত করা। ভিজুয়্যাল বেসিক.নেট প্রথমটির জন্য যা দিয়েছে তার নাম Provider এবং দ্বিতীয়টির জন্য দিয়েছে Data Source। সুতরাং, এবার আমরা উপরের লাইনের পরে নিচের ৪টি লাইন লিখবো বোতামের ক্লিক ইভেন্টে। এর প্রথম তিনটি স্ট্রিং ভেরিয়েবল ডিক্লারেশন। বাকি একটি ভেরিয়েবল এসাইনমেন্ট।

```
Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে [দেখুন ৬ষ্ঠ লাইনটি]।
Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
```

লক্ষ করুন, উপরের শেষের লাইনের কোড হতে পারে: dbProvider = "PROVIDER=Microsoft.**ACE.OLEDB.15.0**;" -কোন্‌টি হবে তা নির্ভর করবে আমাদের ভার্সনের ওপর। আমাদের ব্যবহৃত ভার্সনে যদি কাজ হয় না- আপনি দ্বিতীয়টি টেস্ট করে দেখতে পারেন। এবার নিচের লাইনটি লিখুন:

```
FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডাটাবেইজের পুরো প্যাথ নাম।
```

এবার নিচের লাইনটি লিখুন:

```
dbSource = "Data Source=" & FullDatabasePath 'dbSource মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
```

এ পর্যন্ত কোডিংয়ে কি লিখা হলো তা নিচে তুলে ধরছি। লক্ষ করুন ডিবাগের জন্য ব্যবহৃত মেসেজবক্স ফাংশনটি। এটা অবশ্য পরে কমেন্ট দ্বারা বা মুছে অকেজো করা হবে।

```
Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে [দেখুন ৬ষ্ঠ লাইনটি]।
Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
```

১৫তম পরিচ্ছেদ: ***Database Programming Made Simple***

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডেক্সটপে থাকা ডটাবেইজের পুরো প্যাথ নাম।
dbSource = "Data Source=" & FullDatabasePath 'dbSource মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
'MsgBox(dbSource) 'ডিবাগ কারর জন্য এই লাইন। এটি থাকবে না।
```

এ পর্যন্ত কোড দ্বারা আমরা dbSource স্ট্রিং ভেরিয়েবলে কি হবে তা নিশ্চিত করেছি। অর্থাৎ ডাটা সূত্রের প্যাথ নাম সংরক্ষণ করেছি। কোডের প্রথম লাইনে আমরা যে অবজেক্ট ভেরিয়েবল ডিক্লায়ার করেছিলাম এখন তাকে কাজে লাগাতে হবে। এজন্য একে দু'টি তথ্য দেওয়া প্রয়োজন: ১. ডাটাবেইজ প্রভাইডার ও ২. ডাটাসূত্র। আমরা উভয়টিই নির্দিষ্ট করে নিয়েছি। অর্থাৎ ডাটাবেইজ প্রভাইডার হলো: "PROVIDER=Microsoft.ACE.OLEDB.12.0;" [dbProvider] এবং ডাটাসূত্র হলো "Data Source = " & FullDatabasePath [dbSource]। আর এ উভয় স্ট্রিং মিলেই তৈরি হয় কানেকশন স্ট্রিংটি। সুতরাং আমরা এবার এ লাইনটি কোডে লিখতে পারি:

con.ConnectionString = dbProvider & dbSource

অবশেষে আমাদের কানেকশন স্ট্রিং প্রস্তুত। এখন অতি সহজেই আমাদের ডাটাবেইজ খুলতে পারি। এজন্য একটি মাত্র ছোট্ট কোড লাইনের প্রয়োজন:

con.Open( ) -এ কোডটুকু হলো OleDb.OleDbConnection অবজেক্টের একটি ফাংশন।

ডাটাবেইজ খোলার পর যাকিছু ইচ্ছে তা সেরে একে আবার ক্লোজ করতে হবে। এটা খুব গুরুত্বপূর্ণ ব্যাপার। ক্লোজ করার জন্যও আমরা আরেকটি সহজ ফাংশন ব্যবহার করে থাকি:

con.Close( ) -ব্যস! ডাটাবেইজ খোলা ও বন্ধ করার কোড শেষ হলো।

এ পর্যন্ত কোডে যা হয়েছে তাহলো:

```
Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে [দেখুন ৬ষ্ঠ লাইনটি]।
Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডেক্সটপে থাকা ডটাবেইজের পুরো প্যাথ নাম।
dbSource = "Data Source=" & FullDatabasePath 'dbSource মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
'MsgBox(dbSource) 'ডিবাগ কারর জন্য এই লাইন। এটি থাকবে না।
con.Open( ) 'ডাটাবেইজ খোলা।
con.Close( ) 'ডাটাবেইজ বন্ধ করা।
```

আমরা কাঙ্ক্ষিত গন্তব্যের অর্ধেক পথে এসে গেছি। এবার আরো অগ্রসর হতে হবে- অর্থাৎ, ডাটাবেইজ প্রদর্শনের দরকার এবং বিভিন্ন তথ্য দেখা ও প্রয়োজনে আপডেট করা। পরের পৃষ্ঠাসমূহে আমরা এ সম্পর্কিত কোডগুলো লিখবো। শিক্ষার্থীরা ধৈর্যসহ আমার সাথেই থাকুন এবং স্টাডি করুন।

ডাটাসেট [DataSet] ও ডাটা এডাপটার [Data Adapter]

ডাটা বা তথ্য ডাটাবেজ থেকে কম্পিউটারের মেমোরিতে সংরক্ষণ করার পদ্ধতি হিসাবে আমরা যে সিস্টেম ফাংশন ব্যবহার করি তাকে ADO.Net নামকরণ করেছে DataSet । এটা টেবিলের তথ্যাদির একটি কপি মেমোরিতে হোল্ড করে রাখে। এটা এমন কিছু নয় যা আমাদের ফর্মে লিখে দেখানো যাবে। তবে মেমোরির মধ্যেও এটি কাল্পনিক টেবিল হিসাবে থাকে- অর্থাৎ কলাম ও রো-তে সব তথ্য সংরক্ষিত হয়। একে ডাটাবেইজের ভাষায় ফিল্ড সিস্টেম বলে। তবে প্রশ্ন হচ্ছে DataSet ও Connection Object কি একে অন্যকে দেখতে পারে- বা এমনকি যোগাযোগ করতে সক্ষম?

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

এর জবাব হলো, না। এরা আলাদা। সুতরাং এদের মধ্যকার কমিউনিকেশন ফাঁকটি কোনো উপায়ে পূরণ করতে হবে। আর এ কাজের জন্যই তৈরি হয়েছে Data Adapter।

Data Adapter হলো DataSet ও Connection Object-এর মধ্যকার যোগাযোগ সৃষ্টিকারী মাধ্যম। ডাটা এডাপটার আপনার কানেকশন অবজেক্টের সঙ্গে যোগাযোগ করে এবং যে অনুসন্ধান আপনি সেট করেন তা কার্যকর করে দেয়। ফলাফল ডাটা সেটে সংরক্ষিত হয়। ডাটা সেট ও ডাটা এডাপটার উভয়টিই অবজেক্ট টাইপ। সুতরাং এদেরকে সেট করতে আমরা ব্যবহার করি নিচের ডিক্লারেশগুলোর মতো:

```
Dim ds As New DataSet
Dim da As OleDb.OleDbDataAdapter
```

এরপর:

```
da = New OleDb.OleDbDataAdapter(sql, con)
```

এ কোডগুলোর ব্যাখ্যা এরূপ: DataSet একটি অবজেক্ট। তাই একে New কিওয়ার্ড দ্বারা ডিক্লারেশন করেছি। এই অবজেক্ট বা বস্তুটি সিস্টেম অবজেক্ট। আর সিস্টেম অবজেক্ট ব্যবহার করতে হলে প্রথমে অবজেক্টের একটি ইন্সট্যান্ট [বা কপি] New কিওয়ার্ড দ্বারা ডিক্লায়ার করতে হয়। অপরদিকে DataAdapter হচ্ছে OLEDB অবজেক্টের একটি প্রোপার্টি বা বৈশিষ্ট্য। সুতরাং আমাদেরকে . [ডট] ব্যবহার করতে হয়েছে ইঙ্গিত [বা পয়েন্ট] করার উদ্দেশ্যে। অন্যকথায় da নামক ভেরিয়েবলে এই অবজেক্টটি পাস [pass] করছি। এই ভেরিয়েবল এখন DataAdapter -এর একটি রেফারেন্স হোল্ড করবে।

যাক, দ্বিতীয় লাইন DataAdapter এর রেফারেন্স হোল্ড করবে আর তৃতীয়টি সৃষ্টি করবে একটি নতুন DataAdapter অবজেক্ট। আমরা এর ব্র্যাকেট দু'টোর মধ্যে দু'টি কোড লিখেছি: sql, con। প্রথমটি হলো Structured Query Language- এর একটি কমান্ডের জন্য সৃষ্ট ভেরিয়েবল এবং দ্বিতীয়টি হলো আমাদের কানেকশন অবজেক্ট। এই অবজেক্টটি আমরা ইতোমধ্যে ডিক্লায়ার করেছি। এবার আমাদের সৃষ্ট ভেরিয়েবল sql-কে সেট করতে হবে। একটু পরই সেখানে আসছি। প্রথমে আবারো উল্লেখ করা জরুরী যে: DataAdapter এর কাজ হলো- কানেকশন অবজেক্ট ও ডাটাসেটের মধ্যে কমিউনিকেশন এজেন্ট হিসাবে কাজ করা।

ডাটাবেইজ প্রোগ্রামিংয়ের ভাষা: Structured Query Language [SQL]

ইতোমধ্যে আমরা বেশ কয়েকবার sql [উচ্চারণ, সিকুয়েল] শব্দটি নিয়ে কথা বলেছি। তবে এর ব্যাখ্যা দিই নি। বাস্তবে এটা একটি ডাটাবেইজ প্রোগ্রামিং ভাষার নাম। এ ভাষা সম্পর্কে অন্তত কিছুটা জ্ঞান না থাকলে সত্যিকার অর্থে কোডের মাধ্যমে যে কোন ডাটাবেইজ নিয়ে কাজ করা আদৌ সম্ভব নয়। আমরা এখানে Access ডাটাবেইজ নিয়ে কাজ করছি। সিকুয়েল দ্বারা এরূপ ডাটাবেইজকে নিয়ন্ত্রণ করা যায়। তবে আরো কিছু ডাটাবেইজ আছে যেগুলোর সঙ্গে যোগাযোগ সৃষ্টিতে এই একই ভাষা ব্যবহার করা সম্ভব। এ গ্রন্থে আমরা সেদিকে যাচ্ছি না। শুধু Access ডাটাবেইজের ওপরই আমাদের আলোচনা সীমিত রাখবো। ভাষার মৌলিক নির্দেশনা বেশ সহজ। যেমন:

SELECT * FROM TableName -এই লাইনের অর্থ ডাটাবেইজের টেবিলের পুরো রেকর্ড। লক্ষ করুন **TableName** সিকুয়েলের নির্দেশনা নয়- এটা হচ্ছে যে টেবিল নিয়ে কাজ করবেন তার নাম। যেমন, আমাদের ক্ষেত্রে এটি হবে: Tikana। * অর্থ 'All Records '। সিকুয়েলের সকল কিওয়ার্ড বড়ো অক্ষরে লেখা উত্তম। ফলে আমরা বুঝতে পারবো কোন্‌টি কিওয়ার্ড ও কোন্‌টি নয়। সুতরাং এ লাইনটির অর্থ হলো: **TableName** নামক টেবিল থেকে সকল রেকর্ড সিলেক্ট করো। তবে * -এটা দিয়ে টেবিলের সকল রেকর্ড সিলেক্ট করা ছাড়াও আমরা চাইলে শুধুমাত্র একটি কলাম বা রো সিলেক্ট করতে পারি। যেমন:

```
SELECT Tikana.নাম, Tikana.মোবাইল FROM Tikana
```

আমাদের উদ্দেশ্য কিন্তু পুরো টেবিলটিই সিলেক্ট করা। সুতরাং সিকুয়েলের অন্যান্য কমান্ড সম্পর্কে আর অতিরিক্ত কিছু বলছি না। শিক্ষার্থীরা চাইলে সিকুয়েল বিষয়ের ওপর গ্রন্থাদি পাঠ করতে পারেন। ইন্টারনেটেও এ ভাষা শেখার জন্য

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

অনেক ওয়েভপেইজ আছে। যাক, আমরা এখন তিনটি ডিক্লারেশন ও দু'টি লাইন বোতামের কোডে যুক্ত করতে চাই:

```
'=======নতুন ডিক্লারেশন ==========
    Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
    Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
    Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করে।
'==========================
'======= নতুন কোড ========
    Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
    ' ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
    da = New OleDb.OleDbDataAdapter(Sql, con)
'=====================
```

এখন পর্যন্ত বোতামের কোড হলো:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
    Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে [দেখুন ১০ম লাইনটি]।
    Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
    Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
    '=======নতুন ডিক্লারেশন ==========
    Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
    Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
    Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করে।
    '==========================
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
    ' MsgBox(CurDir)
    FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডাটাবেইজের পুরো প্যাথ নাম।
    dbSource = "Data Source=" & FullDatabasePath 'dbSource মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
    'MsgBox(dbSource) 'ডিবাগ কারর জন্য এই লাইন। এটি থাকবে না।
    con.ConnectionString = dbProvider & dbSource
    con.Open()
    '======= নতুন কোড ========
    Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
    ' ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
    da = New OleDb.OleDbDataAdapter(Sql, con)
    '=====================
    con.Close()
End Sub
```

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

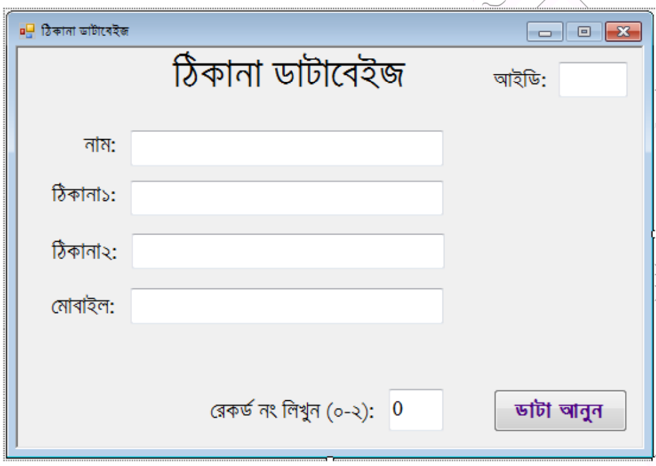
সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

এবার ডাটাসেট পূরণ করার পালা। এজন্য প্রয়োজন আমাদের ডিক্লায়ার করা ডাটা এডাপটার- তথা da। সুতরাং কোড ব্লকের লাইন: da = New OleDb.OleDbDataAdapter(Sql, con) -এর পরই আমাদেরকে নিচের লাইনটি লিখতে হবে।

da.Fill(ds, "TikanaDB") '[illegible]।

আমাদের ডাটা এডাপটার সৃষ্টি হয়ে গেছে। এখন DataSet(ds) -টি আমাদের বানানো TikanaBoi.accdb ডাটাবেইজের **Tikana** নামক টেবিলটি পুরোপুরি হোল্ড করবে। তবে এক সমস্যা তো রয়েই গেলো! কেউ তো তথ্যাদি দেখতে পারবেন না। এ ব্যাপারে আরো কিছু কাজ করতে হবে। আর তা আমরা এক্ষুণি করবো।

টেবিলের তথ্যাদি প্রদর্শনের জন্য প্রয়োজন কয়েকটি টেক্সট্ বাক্স। সুতরাং নিচের চিত্র অনুযায়ী সবক'টি টেক্সট্ বাক্স ও লেবেল ফর্মে নিয়ে যুক্ত করুন।



লক্ষ করুন: চারটি খালি টেক্সট্ বাক্সে থাকবে টেবিলের তথ্যাদি। উপরের ছোট্ট টেক্সট্ বাক্সে থকাবে আইডি নাম্বার। নিচের ছোট্ট টেক্সট্ বাক্সে থাকবে রেকর্ড নাম্বার। যেহেতু রেকর্ড শুরু হয় ০ থেকে তাই এটাই থাকবে ডিফোল্ট হিসাবে। আমাদের কোড এই সংখ্যাটি নিয়ে তথ্য প্রদর্শন করবে। ব্যবহারকারী যখন ১ কিংবা ২ লিখবেন এবং বোতামে টিপ দিবেন তখন এই সংখ্যা মুতাবিক রেকর্ড প্রদর্শন হবে। যাক যে অতিরিক্ত লাইনগুলো বোতামের ক্লিক ইভেন্টে লিখা হবে সেগুলো নিচে তুলে ধরছি। এরপর পুরো কোডটি আমরা তুলে ধরবো।

যেহেতু আমাদের টেবিলে প্রত্যেক রেকর্ডে ৪টি করে রেকর্ড আছে তাই এগুলো প্রদর্শনের জন্য চারটি ফিল্ডের কথাই লিখতে হবে। তবে সর্বপ্রথম আমাদেরকে জানতে হবে, ব্যবহারকারী যে সংখ্যাটি রেকর্ড নং ফিল্ডে লিখেছেন তা সঠিক কি না।

```
'validate record number
If Val(TextBox1.Text) < 0 Or Val(TextBox1.Text) > 2 Then 'ভেলিড সংখ্যা না হলে
    MsgBox("Please enter 0, 1 or 2 only") 'এই মেসেজ প্রদর্শন করে সাব থেকে বেরিয়ে যাওয়া।
    Exit Sub
End If
'==============
    Dim RecNum As Integer = 0 'এটা হোল্ড করবে টেক্সট্ বাক্সের সংখ্যাটি যার দ্বারা সঠিক রো এর তথ্য আনা হবে।
    RecNum = Val(TextBox1.Text) 'ভেরিয়েবলে টেক্সবাক্সের সংখ্যা [০,১ বা ২]
```

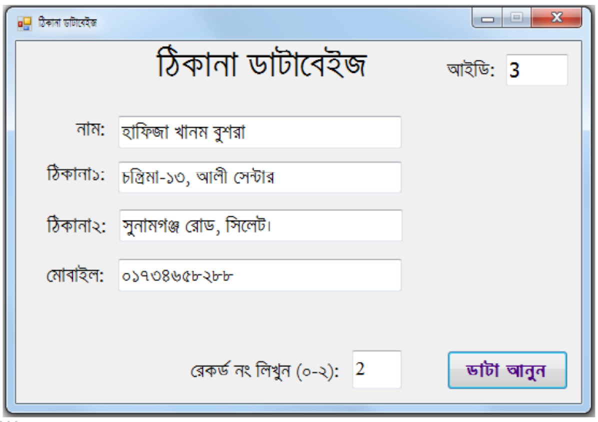
উক্ত কোড ব্লক কিভাবে কাজ করে তা আশারাখি বুঝিয়ে বলার প্রয়োজন নেই। If -- End if ব্লকের কাজ হলো টেক্সট্ বাক্স-১ এর ভেতর ব্যবহারকারী সঠিক সংখ্যা [০,১ বা ২] লিখেছেন কি না। টেস্ট ফেইল হলে একটি মেসেজ বাক্স দেখিয়ে বোতামের ইভেন্ট সাব থেকে বেরিয়ে আসার ব্যবস্থা করা হয়েছে। এরপর আবার সঠিক সংখ্যা লিখলে কিন্তু টেবিল থেকে চারটি ভ্যালু উপরের চারটি টেক্সট্ বাক্সে প্রদর্শিত হবে। লক্ষ করুন, ডিফোল্ট সংখ্যা হিসাবে ০ রেখেছি। টেক্সবাক্স-১ এর Text প্রোপার্টিতে ডিজাইন-টাইমে ০ লিখে দিয়েছি। অপর আরেক ব্যাপার হলো উপরের চারটি টেক্সবাক্সকে রীড-অনলি বাক্সে পরিণত করেছি যাতেকরে কেউ কিছু এতে লিখতে পারেন না। প্রোপার্টিতে যেয়ে টেক্সবাক্সের ReadOnly প্রোপার্টি True করলেই কাজ হয়ে যাবে। যাক এবার আমরা পরের পৃষ্ঠায় পূর্ণ কোডটি তুলে ধরছি। শিক্ষার্থীরা কোডগুলো দেখে দেখে ভালো করে বুঝার চেষ্টা করবেন। অবশ্য কোড লেখার পর আমরা প্রোগ্রামটি রানকালীন একটি জানালার চিত্রও দেখিয়েছি। আমাদের ডাটাবেইজের আইটেমগুলো সঠিকভাবে আসছে কি না তা তো টেস্ট করে দেখা চাই। ঠিক না?

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        'validate record number
        If Val(TextBox1.Text) < 0 Or Val(TextBox1.Text) > 2 Then 'ভেলিড সংখ্যা না হলে
            MsgBox("Please enter 0, 1 or 2 only") 'এই মেসেজ প্রদর্শন করে সাব থেকে বেরিয়ে যাওয়া।
        Exit Sub
        End If
        '==============
        Dim RecNum As Integer = 0 'এটা হোল্ড করবে টেক্সট বাক্সের সংখ্যাটি যার দ্বারা সঠিক রো এর তথ্য আনা হবে।
        RecNum = Val(TextBox1.Text) 'ভেরিয়েবলে টেক্সবাক্সের সয়খ্যা [০,১ বা ২]
        Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
        Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে।
        Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
        Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
        Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
        Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
        Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করে।
        '===========================
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
        FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডাটাবেইজের পুরো প্যাথ নাম।
        dbSource = "Data Source=" & FullDatabasePath 'dbSource এর মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
        'MsgBox(dbSource) 'ডিবাগ কারর জন্য এই লাইন। এটি থাকবে না।
        con.ConnectionString = dbProvider & dbSource
        con.Open() 'ডাটাবেইজটি খোলার নিরর্দশ।
        Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
        'ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
        da = New OleDb.OleDbDataAdapter(Sql, con)
        da.Fill(ds, "TikanaDB") 'দ্বিতীয় প্যারামিটার যে কোনো কিছু হতে পারে।
        IDtext.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(0) 'আইডি
        NameText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(1) 'নাম
        AddressText1.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(2) 'ঠিকান-১
        AddressText2.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(3) 'ঠিকানা-২
        MobileText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(4) 'মোবাইল
        '======================
        con.Close() 'ডাটাবেইজটি বন্ধের নির্দেশ।
    End Sub
End Class
```

ডানের চিত্রে ঠিকানা ডাটাবেইজ প্রোগ্রামটি রানকালীন জানালার একটি দৃশ্য দেখাচ্ছে। লক্ষ করুন, রেকর্ড দেখার জন্য আমাদেরকে একটি ইন্টেজার [০,১,২] প্রবেশ করতে হবে। কিন্তু আমরা যখন উইজার্ড দিয়ে একই ডাটাবেইজ তৈরি করেছিলাম, তখন ন্যাভিগেট করার জন্য দু'টি বোতাম ছিলো। ওগুলোতে ক্লিক করে করে আমরা এক রেকর্ড থেকে অন্যটিতে যাই। আর এটাই স্বাভাবিক। সুতরাং কোড দ্বারা সৃষ্ট আমাদের ঠিকানা ডাটাবেইজও কি অনুরূপ প্রাকটিক্যাল বানাতে পরি? হ্যাঁ, পারি। সুতরাং আসুন, এবার ন্যাভিগেশন কোডগুলো কিভাবে লিখতে হবে তা জেনে নিই। প্রথমে ফর্মের ডিজাইনে কিছু পরিবর্তন আনতে হবে। নিচের চিত্রে সেটা তুলে ধরেছি।

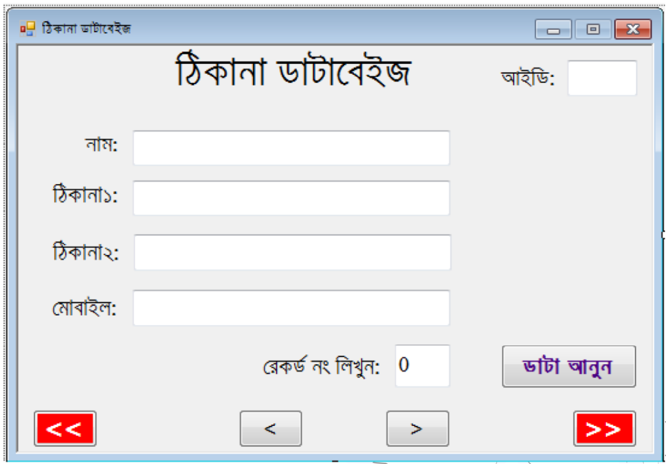


এখানে আমরা যা করেছি তাহলো, চারটি ন্যাভিগেশন বোতাম যুক্ত করেছি। সর্বাধিক ডানের ও বামের বোতাম দু'টো হলো যথাক্রমে

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

শেষের রেকর্ড ও প্রথম রেকর্ড দেখার বোতাম। মাঝের দু'টো প্রদর্শিত রেকর্ড থেকে পরেরটা ও আগেরটায় যাওয়ার বোতাম। কোডের সুবিধার্তে প্রতিটি বোতামের নামকরণ করেছি এভাবে:



সর্বাধিক ডানের বোতাম: btnLast।
সর্বাধিক বামের বোতাম: btnFirst।
পরের রেকর্ডের বোতাম: btnNext।
আগের রেকর্ডের বোতাম: btnPrev।

আমাদের ইচ্ছে ঠিকানা ডাটাবেইজটি ব্যবহারবান্ধব হবে। সুতরাং আমরা চাই রেকর্ড দেখার জন্য রেকর্ড সংখ্যা দ্বারা যেরূপ রেকর্ডের তথ্য নিয়ে আসা যায় সেরূপ যাতে ন্যাভিগেশন বোতাম দ্বারাও রেকর্ড দেখার সুযোগ থাকে। এছাড়া প্রথম ও শেষ রেকর্ড দেখারও সহজ উপায় থাকতে হবে। এ কারণেই 'ডাটা আনুন' বোতাম এবং সংখ্যা প্রবেশের ফিল্ডটি রেখে দিয়েছি।

ন্যাভিগেশনের জন্য নতুন কোড লিখতে হবে। তবে আমরা ইতোমধ্যে যাকিছু কোড করেছি তার বেশিরভাগই ফাইনাল কোডে থাকবে। নিচে এবার পুরো কোডটি তুলে ধরেছি। এরপর অতিরিক্ত কোড কিভাবে কাজ করে তা একে একে বুঝিয়ে দেবো। এই ব্যাখ্যা শেষে Database Programming এর মতো বিরাট বিষয়ের ওপর আমাদের প্রাথমিক অনুসন্ধানমূলক এ পরিচ্ছেদের ইতি ঘটবে। শিক্ষার্থীরা এ থেকেই যা শিখেছেন, সাধারণ কাজের জন্য তা-ই যথেষ্ট হবে। ইন্টারনেটে অনেক স্যাম্পুল ডাটাবেইজ এ্যাপলিকেশন পাওয়া যায়। যারা এ বিষয়ে আরো গভীরে যেতে ইচ্ছুক তাদের প্রতি আমার উপদেশ হলো ইন্টারনেটের এসব স্যাম্পুল ডাউনলোড করে কোড দেখুন ও শিখুন।

ঠিকানা ডাটাবেইজের এ পর্যন্ত পূর্ণাঙ্গ কোড

```
Public Class Form1
    'গ্লোবাল ডিক্লারেশন
    Dim RecNum As Integer = 0 'এটা হোল্ড করবে টেক্সট বাক্সের সংখ্যাটি যার দ্বারা সঠিক রো এর তথ্য আনা হবে।
    Dim NumRec As Integer 'এটা হোল্ড করবে টেবিলের মোট রেকর্ড নাম্বার।
    'এটা হলো ডাটা আনুন বোতামের সাবরুটিন। এটা কল করে GetAndFillRecord সাব।
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        GetAndFillRecord() 'সাব কল।
    End Sub
    'এই সাব ন্যাভিগেশন বোতামগুলো সক্রিয় করে। প্রাথমিকভাবে ওগুলো অক্রিয়।
    Private Sub EnableNavigationBtn()
        btnFirst.Enabled = True : btnLast.Enabled = True
        btnNext.Enabled = True : btnPrev.Enabled = True
   End Sub
    'এ ফাংশনের কাজ হলো টেক্সট বাক্সে সঠিক সংখ্যা আছে কি না, তা দেখা। রিটার্ন করবে বুলিয়ান ভ্যালু [সত্য বা মিথ্যা]।
    Private Function ValidateEntryNum()
        If Val(TextBox1.Text) < 0 Or Val(TextBox1.Text) > NumRec - 1 Then 'ভেলিড সংখ্যা না হলে
            'এই ফাংশন কলের পূর্বে অন্তত একবার ডাটাবেইজ খোলা হবে। ফলে NumRec এর মান কি তা-ও জানা থাকবে
        MsgBox("There are 0 to " & NumRec - 1 & " records only!") 'এই মেসেজ প্রদর্শন করে ফলস ভ্যালু রিটার্ন করা।
            Return False
        End If
        Return True 'গ্রহণযোগ্য সংখ্যা- তাই রিটার্ন ট্রু।
    End Function
    'রেকর্ড থেকে তথ্য এনে ফিল্ডে দেখানো হলো এ সাবের কাজ
    Private Sub GetAndFillRecord()
```

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
        RecNum = Val(TextBox1.Text) 'ভেরিয়েবলে টেক্সবাক্সের সংখ্যা। প্রথমবার এর মান ০।
        Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
        Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে।
        Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
        Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
        Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
        Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
        Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করে।
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
        FullDatabasePath = "C:\Users\htci\Desktop\LearningDotNet\MyDatabase\TikanaBoi.accdb" 'ডাটাবেইজের পুরো প্যাথ নাম।
        dbSource = "Data Source=" & FullDatabasePath 'dbSource এর মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
        con.ConnectionString = dbProvider & dbSource 'কানেকশন স্ট্রিং পূর্ণ হলো।
        con.Open() 'ডাটাবেইজ খোলার নির্দেশ।
        'ক টি রেকর্ড আছে তা জানা। ভেরিয়েবল cmd একটি সিকুয়েল কমান্ড হোল্ড করবে।
        Dim cmd As New OleDb.OleDbCommand("Select COUNT(*) FROM Tikana", con)
        NumRec = cmd.ExecuteScalar() 'সিস্টেম ফাংশন দ্বারা রেকর্ড সংখ্যা আনা।
        'ফাংশন কল করে রেকর্ড সংখ্যা সঠিক আছে কি না দেখা।
        If ValidateEntryNum() = False Then
            con.Close() 'ভুল হলে ডাটাবেইজ বন্ধ করে এক্সিট করা
            Exit Sub
        End If
        Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
        'ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
        da = New OleDb.OleDbDataAdapter(Sql, con)
        da.Fill(ds, "TikanaDB") 'দ্বিতীয় প্যারামিটার যে কোনো কিছু হতে পারে।
        IDText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(0) 'আইডি
        NameText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(1) 'নাম
        AddressText1.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(2) 'ঠিকান-১
        AddressText2.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(3) 'ঠিকানা-২
        MobileText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(4) 'মোবাইল
        EnableNavigationBtn() 'ডাটাবেইজ লোড হয়েছে তাই চারটি ন্যাভিগেশন বোতাম সক্রিয় করতে হবে।
        '======================
        con.Close() 'ডাটাবেইজ ক্লোজ করা- এটা জরুরী।
    End Sub
    'এটা প্রথম রেকর্ড দেখার বোতামের সাবরুটিরন।
    Private Sub btnFirst_Click(sender As Object, e As EventArgs) Handles btnFirst.Click
        TextBox1.Text = "0" 'ট্যাক্সট বাক্সে প্রথম রেকর্ড ইনডেক্স সংখ্যা লিখে দেওয়া।
        GetAndFillRecord() 'উপলের রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
    End Sub
    'এটা সর্বশেষ রেকর্ড দেখার বেতামের সাবরুটিন।
    Private Sub btnLast_Click(sender As Object, e As EventArgs) Handles btnLast.Click
        TextBox1.Text = NumRec - 1 'ইনডেক্স শুরু হয় ০ থেকে তাই মোট রেকর্ড থেকে ১ কমাতে হবে।
        GetAndFillRecord() 'রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
    End Sub
    'এটা পরের রেকর্ড দেখার বোতামের সাবরুটিন।
    Private Sub btnNext_Click(sender As Object, e As EventArgs) Handles btnNext.Click
        If ValidateEntryNum() <> False Then 'প্রথমে ভেলিড সংখ্যা আছে কি না দেখতে হবে।
            If TextBox1.Text >= NumRec Then 'ভেলিড সংখ্যা সত্য হলেও, সর্বশেষ রেকর্ডে আসা হয়েছে কিনা দেখতে হবে।
                Beep() 'সর্বশেষ রেকর্ড এটি- সিস্টেম সাউন্ড প্লে করে সাব থেকে বেরিয়ে যাওয়া।
                Exit Sub
            End If
```

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
        Dim entryNum As Integer = Val(TextBox1.Text) + 1 'টেক্স বাক্সের সংখ্যা ১ বাড়ানো।
            If entryNum >= NumRec Then 'আবার চেক করা দরকার, এ নতুন সংখ্যা রেকর্ড সংখ্যা থেকে বেশি কি না।
                Beep() 'বেশি হলে বিপ বাজিয়ে আউট!
                Exit Sub
            End If
            TextBox1.Text = entryNum 'নতুন নাম্বার টেক্স বাক্সে নেওয়া ও...
            GetAndFillRecord() 'প্রদর্শন সাব কল করা।
        End If
    End Sub
    'এটা আগের রেকর্ড দেখার বোতামের সাবরুটিন।
    Private Sub btnPrev_Click(sender As Object, e As EventArgs) Handles btnPrev.Click
        If ValidateEntryNum() <> False Then 'প্রথমে ভেলিড সংখ্যা আছে কি না দেখতে হবে।
            If TextBox1.Text <= 0 Then 'ভেলিড সংখ্যা সত্য হলেও, প্রথম রেকর্ডে আসা হয়েছে কিনা দেখতে হবে।
            Beep() 'প্রথম রেকর্ড এটি- সিস্টেম সাউন্ড প্লে করে সাব থেকে বেরিয়ে যাওয়া।
            Exit Sub
        End If
        Dim entryNum As Integer = Val(TextBox1.Text) - 1 'টেক্স বাক্সের সংখ্যা ১ কামনো।
        If entryNum < 0 Then 'আবার চেক করা দরকার, এ নতুন সংখ্যা শূন্য বা প্রথম রেকর্ড সংখ্যা থেকে কম কি না।
            Beep() 'কম হলে বিপ বাজিয়ে আউট!
            Exit Sub
        End If
        TextBox1.Text = entryNum 'নতুন নাম্বার টেক্স বাক্সে নেওয়া ও...
        GetAndFillRecord() 'প্রদর্শন সাব কল করা।
        End If
    End Sub
End Class
```

ঠিকানা ডাটাবেইজের পূর্ণাঙ্গ কোড তুলে ধরলাম। আমরা একে একে প্রতিটি সাব নিয়ে এবার বিস্তারিত ব্যাখ্যা তুলে ধরবো। তবে ইতোমধ্যে যেসব ব্যাপারে আলোচনা হয়ে গেছে তার পুনরাবৃত্তি করবো না অবশ্যই।
আমাদের কোডে সর্বমোট ৭টি সাবরুটিন ও ১টি ফাংশন আছে। প্রথমে এগুলোর নামের একটি তালিকা তুলে ধরছি।
১. Private Sub Button1_Click। ২. Private Sub EnableNavigationBtn। ৩. Private Sub GetAndFillRecord। ৪. Private Sub btnFirst_Click। ৫. Private Sub btnLast_Click। ৬. Private Sub btnNext_Click। ৭. Private Sub btnPrev_Click।
একটি মাত্র ফাংশন: Private Function ValidateEntryNum।

আমি শিক্ষার্থীদের সুবিধার জন্য মূল কোডে খুব বেশি করে কমেন্ট করেছি। এতে অনেক ব্যাখ্যা এসেছে। বাস্তবে যে কোনো কোডেই এরূপ ব্যাখ্যা রাখা জরুরী। পরবর্তীতে ডিবাগ [সঠিককরণ] ক্রিয়া অনেকটা সহজ হয়। আর মনে রাখা দরকার, কোনো কোনো কোড শত শত বা এমনকি হাজার হাজার লাইন পর্যন্ত হতে পারে। সুতরাং শিক্ষার্থীরা কমেন্ট করার অভ্যেস করে নেবেন। মনে রাখবেন যে কোনো লাইনের পূর্বে এই সিঙ্গুল কোট মার্ক ['] থাকলে সেটা কমেন্ট হয়ে যায়। কম্পাইল বা রান করার সময় এ লাইন উপেক্ষা করা হবে। যাক এবার ৭টি সাবরুটিন ও ১টি ফাংশনের ব্যাখ্যা একে একে তুলে ধরার প্রয়াস পাচ্ছি।
আমরা দু'টি গ্লোবাল ভেরিয়েবল ফর্মের ক্লাসের একেবারে উপরেই ডিক্লায়ার করেছি। আমাদের উদ্দেশ্য এগুলো যে কোনো সাবে প্রাপ্য হওয়া। এ গ্লোবালের প্রথমটির নাম: RecNum। আর দ্বিতীয়টির নাম: NumRec। প্রথমটিতে থাকবে ব্যবহারকারীর প্রবেশকৃত সংখ্যা- অর্থাৎ TextBox1 -এর সংখ্যাটি। আর দ্বিতীয়টিতে থাকবে আমাদের ঠিকানা ডাটাবেইজের টেবিলের রেকর্ড সংখ্যা। আমরা চাই রেকর্ড বৃদ্ধি কিংবা কমলেও যাতে এ্যাপলিকেশন কাজ করে। কোড ব্যাখ্যার পরই আমরা কিছু রেকর্ড বাড়াবো এবং প্রোগ্রাম রান করে দেখবো সবকিছু ঠিকমতো কাজ করছে কি না।

১. Private Sub Button1_Click: এই সাবরুটিন দ্বারা আমরা রেকর্ড প্রদর্শন করবো। এটা মূল সাবরুটিন

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

GetAndFillRecord কল করে মাত্র।

২. Private Sub EnableNavigationBtn: এটার কাজ চারটি ন্যাভিগেশন বোতাম সক্রিয় করা। প্রথমে প্রোগ্রাম চলাকালে এই বোতামগুলো অক্রিয় রাখা থাকবে। এর কারণ হলে, তখনো ডাটাবেইজ খুলবে না। সুতরাং এগুলো সক্রিয় থাকার প্রয়োজন নেই। এই সাব ডাটাবেইজ খোলার পরই ডাকা হবে।

৩. Private Sub GetAndFillRecord: এটাই আমাদের মূল সাবরুটিন যার মাধ্যমে ডাটাবেইজ খোলা ও প্রদর্শিত হয়। এ সাব সম্পর্কে আগেই আমরা বেশ আলোচনা করেছি। তবে কিছু কোড নতুনভাবে যুক্ত হয়েছে। সুতরাং ওগুলো সম্পর্কে কিছু ব্যাখ্যার প্রয়োজন।

প্রথম লাইনটি দ্বারা আমরা গ্লোবাল ভেরিয়েবল RecNum -এ মান যুক্ত করেছি। লক্ষ করুন, এ মানটি হলো টেক্সবাক্স-১ এর মধ্যে যা আছে তাই। আর প্রথমে এ সংখ্যা হবে ০। অবশ্য পরবর্তিতে এটা ভিন্ন হতে পারে। এরপর যে কোডটি এখানে যুক্ত হয়েছে তাহলো con.Open() -এর পরের দু'টো লাইন এবং সবশেষে একটি লাইন। যথা:

```
'ক টি রেকর্ড আছে তা জানা। ভেরিয়েবল cmd একটি সিকুয়েল কমান্ড হোল্ড করবে।
Dim cmd As New OleDb.OleDbCommand("Select COUNT(*) FROM Tikana", con)
NumRec = cmd.ExecuteScalar() 'সিস্টেম ফাংশন দ্বারা রেকর্ড সংখ্যা আনা।
TextBox2.Text = NumRec 'রেকর্ড সংখ্যা টেক্স ফিল্ডে দেখানো।
```

লক্ষ করুন, উপরের কমেন্ট লাইনটি রেখে দিয়েছি যাতে শিক্ষার্থীরা ব্যাপারটি সহজে বুঝতে পারেন। প্রথম লাইনটি হলো **OleDb** এর একটি সিস্টেম অবজেক্ট ডিক্লারেশন। অবজেক্টটির একটি ইন্সট্যান্ট (বা কপি) **cmd** নামকরণে নতুনভাবে নিয়ে এসেছি। এর দুটি প্যারামিটার আছে: ১. সিকুয়েল কমান্ড "Select COUNT(*) FROM Tikana" এবং ২ con নামক কানেকশন অবজেক্ট। লক্ষ করুন প্রথম প্যারামিটারের কমান্ডে COUNT শব্দটি আছে। এর অর্থ হলো ঠিকানা টেবিলের মধ্যে ক'টি রেকর্ড আছে তা গণনা করা। দ্বিতীয় লাইন দ্বারা নির্দেশ দেওয়া হচ্ছে কমান্ড অবজেক্ট থেকে সংখ্যা নিয়ে এসে ভেরিয়েবল **NumRec** -তে রাখো। আর এ কাজের জন্য একটি সিকুয়েল নির্দেশনা তথা **ExecuteScalar** ব্যবহৃত হয়েছে। শিক্ষার্থীদের এ পর্যায়ে ঠিক কিভাবে এই কমান্ডগুলো কাজ করে তা জানার দরকার নেই। শুধুমাত্র এটাই শিখে নিতে হবে যে উপরোক্ত দু'টি লাইন দ্বারা যে কোনো ডাটাবেইজের উল্লেখিত টেবিলে ক'টি রো বা রেকর্ড আছে তা জানা যায়। আর মনে রাখার বিষয় যে, ডাটাবেইজ খোলা থাকাবস্থায় এই লাইনগুলো কার্যকরী হবে। এ দু'টি লাইনের একমাত্র উদ্দেশ্য হলো টেবিলে ক'টি রেকর্ড আছে তা জানা। শেষের লাইনটি দ্বারা রেকর্ড সংখ্যাটি টেক্সট বাক্স-২ এ প্রদর্শনের নির্দেশ দেওয়া হয়েছে। অবশ্য ফর্মে একটি লেবেল ও টেক্স বাক্স যুক্ত করতে হবে। এই সাবে আরো যা নতুনভাবে যুক্ত হয়েছে তাহলো:

```
IDText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(0) 'আইডি
```

এবং

```
EnableNavigationBtn() 'ডাটাবেইজ লোড হয়েছে তাই চারটি ন্যাভিগেশন বোতাম সক্রিয় করতে হবে।
```

যেহেতু আমাদের ডাটাবেইজে **ID** নামক একটি কলাম আছে তাই এটাও আইডি ফিল্ডে প্রদর্শনের প্রয়োজন। প্রথম লাইনে এ নির্দেশনাই দেওয়া হয়েছে। আর দ্বিতীয় লাইনের মাধ্যমে আমরা ন্যাভিগেশন বোতাম চারটে সক্রিয় করার জন্য সাবরুটিন কল করেছি।

৪. Private Sub btnFirst_Click। এ বোতামের মাধ্যমে আমরা প্রথম রেকর্ড দেখার ব্যবস্থা করেছি। লক্ষ করুন, অতি সহজে এ কাজটি আঞ্জাম দিতে আমরা প্রথমে টেক্সটবাক্স-১ এ ০ প্রবেশ করেছি এরপর কল করেছি আমাদের মূল সাবটি।

৫. Private Sub btnLast_Click। এ বোতামের মাধ্যমে আমরা সর্বশেষ রেকর্ডটি দেখাতে চাই। সুতরাং কোড প্রায় একই। এ ক্ষেত্রে অবশ্য ক'টি রেকর্ড আছে তা আমাদের জানা নেই। তবে, আমরা NumRec ভেরিয়েবলের মধ্যে এ সংখ্যাটি সংরক্ষণ করেছি। রেকর্ডের [বা রো এর] ইনডেক্স সিস্টেম কিন্তু ০- বেইজড। সুতরাং সর্বশেষ রেকর্ডের ইনডেক্স হবে মোট রেকর্ড মাইনাস ১। এটা মনে রাখতে হবে।

৬. Private Sub btnNext_Click। এটা পরবর্তী রেকর্ডে ন্যাভিগেশনের কোড। তবে কোডটি কিছুটা জটিল। প্রথমে

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

জানতে হবে টেক্সটবাক্সে বর্তমানে যে সংখ্যাটি আছে তা ভেলিড কি না। কারণ, ব্যবহারকারী ইনভেলিড একটি সংখ্যা ঢুকিয়ে এ বোতামে ক্লিক করতে পারেন। ভেলিডিটি পরীক্ষার জন্য আমার `ValidateEntryNum()` টি কল করেছি। এটা কিভাবে কাজ করে একটু পরই বলবো। তবে এর ফলাফল সত্য বা মিথ্যা। প্রথম লাইনে আমরা বলছি রেজাল্ট যদি মিথ্যা না নয় [অর্থাৎ তা সত্য] তাহলে টেস্ট করে দেখতে হবে এ সংখ্যাটি রেকর্ডের সর্বশেষ ইনডেক্স সংখ্যা কি না। এটাও যদি না হয় তাহলেই আমরা পরবর্তী রেকর্ড দেখতে পারি। এ টেস্টের জন্য আমরা >= NumRec -এই কোডাংশ ব্যবহার করেছি। বলা হচ্ছে মোট রেকর্ড সংখ্যার সমান বা বেশি আছে কি? উভয়টির অর্থ হলো শেষ রেকর্ড প্রদর্শিত হচ্ছে। সুতরাং বিপ বাজিয়ে বের হও! অন্যথায়, পরবর্তী রেকর্ডে যাওয়া যায়।

এবার এই কোড লাইন: "`Dim entryNum As Integer = Val(TextBox1.Text) + 1 'টেক্স বাক্সের সংখ্যা ১ বাড়ানো।`" দ্বারা এনট্রি নাম্বার বাড়িয়েছি পরের রেকর্ডে যাওয়ার জন্য- কিন্তু! এখানেও আরেকটি টেস্টের দরকার। যদি নতুন নাম্বার রেকর্ডের সর্বশেষ ইনড্রেক সংখ্যা থেকেও বড়ো হয়ে যায়? সুতরাং দ্বিতীয় টেস্টও পাস করতে হবে। এবার পরের রেকর্ডে যাওয়ার নির্দেশ দেওয়া যায়। এ কাজে প্রথমে টেক্সট বাক্সে নতুন সংখ্যা দিয়েছি এরপর কল করেছি আমাদের মূল সাবটি।

৭. `Private Sub btnPrev_Click`। এটাও এইমাত্র ব্যাখ্যাত সাবের মতো কাজ করে। এ ক্ষেত্রে অবশ্য দেখতে হবে রেকর্ড সংখ্যাটি ০ থেকে নিচে হয়ে যায় কি না- এই যা।

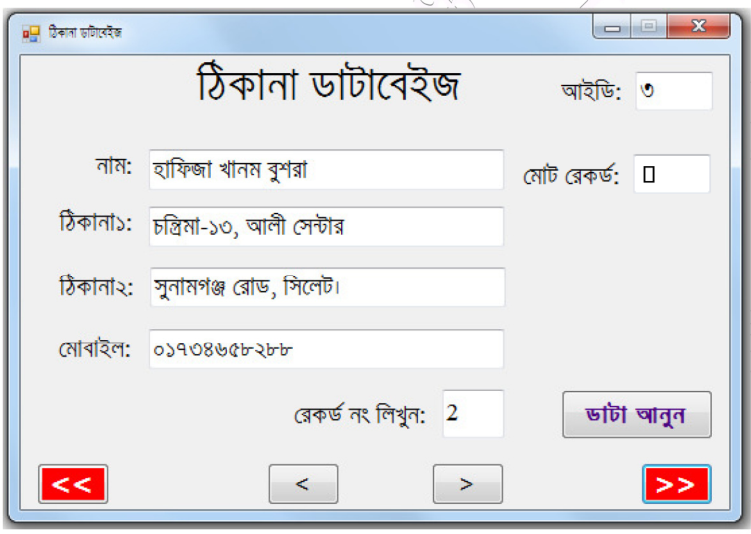
৮. `Private Function ValidateEntryNum`। সবশেষে এই ফাংশনটি কিভাবে কাজ করে তা একটু জেনে নিলে ভালো হয়। ফাংশনের নামেই বুঝা যায় এর কাজ হলো টেবিলের রো বা রেকর্ডের ইনডেক্স সংখ্যা সঠিক কি না যাচাই করে 'সত্য' বা 'মিথ্যা' তথা বুলিয়ান রেজাল্ট ফেরৎ পাঠানো। মনে রাখা প্রয়োজন, আমরা সর্বদাই টেক্সবাক্স-১ এর সংখ্যা নিয়ে এখানে কাজ করছি। এমনকি ন্যাভিগেশন বোতামগুলোর ক্রিয়ার সময়ও এই টেক্সবাক্সে ইনডেক্স সংখ্যা নিয়ে কোড রান করেছি। সুতরাং দেখতে হবে সংখ্যাটি ০ থেকে ইনডেক্স সংখ্যা পর্যন্ত যে কোনো একটি আছে কি না। না থাকলে ফাংশন একটি মেসেজ প্রদর্শন করে False রিটার্ন করবে। অন্যথায় রিটার্ন করবে True। লক্ষ করুন মেসেজ বাক্সে মোট ক'টি ইনডেক্স রেকর্ড আছে তা প্রদর্শিত হবে।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

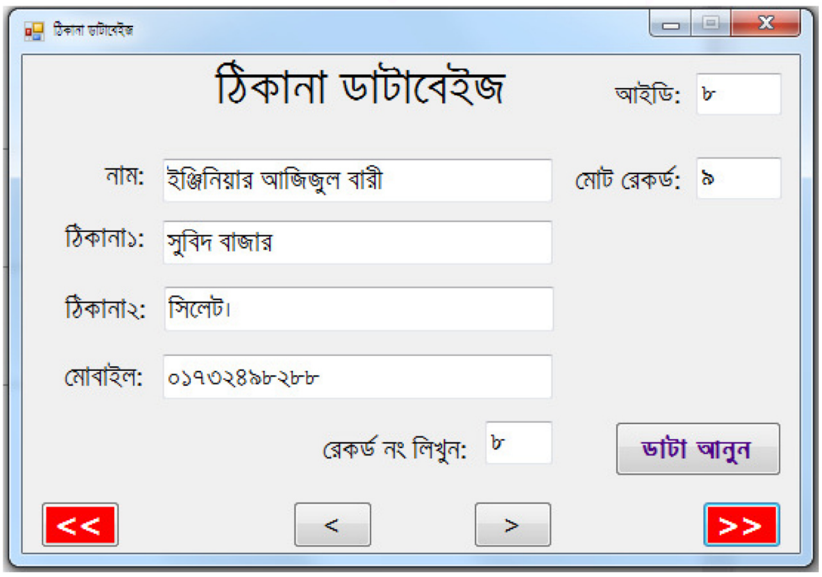
আলহামদুলিল্লাহ! ডাটাবেইজ প্রোগ্রামিংয়ের বেসিক কিছু কোর্ডিং সম্পর্কে আমরা অবগত হলাম। যে দু'টি ডাটাবেইজ এ্যাপলিকেশন আমরা তৈরি করেছি তা একেবারে সাধারণ নয়! এগুলো কাজে লাগানো যাবে। সবচেয়ে বড়ো কথা হলো আমাদের ডাটাবেইজ সম্পূর্ণ **বাংলা!**

এখানে একটি ব্যাপার উল্লেখ করতে হয়, প্রথমটির মতো দ্বিতীয়টিতে আমরা ন্যাভিগেশন কোডও দিয়েছি। তবে দু'টি বিষয় ছাড়া- মুছে ফেলা ও নতুন রেকর্ড তৈরি করা, এ দু'টো সুবিধা নেই। এগুলোর কোডিং বেশ জটিল তাই সেদিকে আপাতত যাচ্ছি না। এ দু'টো সুবিধাসহ ডাটাবেইজ চাইলে আমরা সহজেই ভিজুয়্যাল স্টুটিও ব্যবহার করে উইজার্ড দিয়ে ডাটাবেইজ বানাতে পারি- যেমনটি আমরা প্রথমে করেছি। তবে কোড দ্বারা কিভাবে সম্ভব তা-ও একটু পরে আমারা শিখে নেবো। এ পর্যন্ত যা হয়েছে তা পরীক্ষা করার পালা। আমরা ডাটাবেইজে প্রথম প্রোগ্রাম রান করে আরো কয়েকটি রো বা রেকর্ড যুক্ত করবো। এরপর এটি দ্বিতীয়টি দ্বারা প্রদর্শন করে দেখবো সব ঠিকমতো দেখা যায় কি না- কেমন?

টেস্ট



ঠিকানা ডাটাবেইজের রেকর্ড আগের মতো [৪টি] থাকাবস্থায় এ্যাপলিকেশন রান করে সবগুলো বোতাম পরীক্ষা করেছি। উপরে এর একটি চিত্র দেখুন।



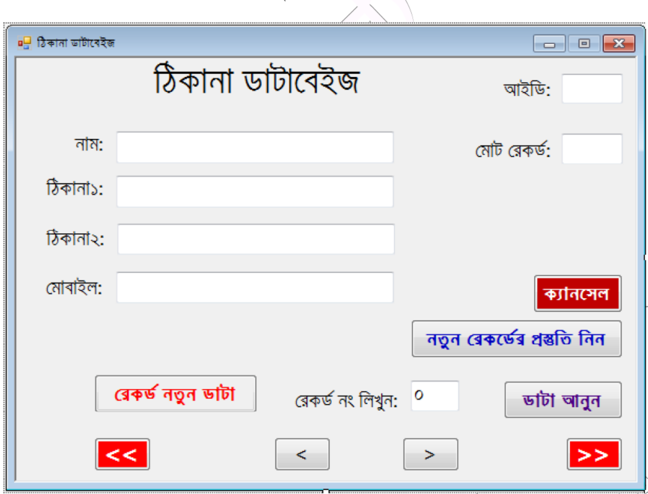
ঠিকানা ডাটাবেইজের রেকর্ডে আরো কিছু [৫টি] নতুন ডাটা যুক্ত করার পর এ্যাপলিকেশন রান করে সবগুলো বোতাম পরীক্ষা করেছি। ডানের চিত্রে আমার সর্বশেষ [৯নং] রেকর্ডটি দেখতে পাচ্ছি।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

ঠিকানা ডাটাবেইজে নতুন রেকর্ড সৃষ্টির কোড

আমরা এ পর্যন্ত যা করেছি তার দ্বারা রেডিমেইড কোনো এক্সেস ডাটাবেইজের তথ্যাদি প্রদর্শন সম্ভব। তবে সময় সময় তো নতুন রেকর্ড প্রবেশ করার প্রয়োজন দাঁড়াতে পারে। সুতরাং আসুন, দেখে নিই কিভাবে কোডের মাধ্যমে তা সম্ভব।

প্রথমত, একটি নতুন বোতাম ফর্মে যুক্ত করতে হবে। এর নামকরণ করুন □□□□□□□□ □□□ □□□□□! প্রয়োজনে আমরা ফর্মের সাইজ কিছুটা বড়ো করবো। বোমাতের নামকরণ করুন: btnNewRecord।



মনে রাখবেন, বাংলা লিখবেন বিজয়ের ইউনিকোডে। ইউনিকোড আনার শর্টকাট: Ctrl + Alt + V। একই শর্টকাট দ্বারা পুনরায় ইংরেজি লিখার কিবোর্ড আনা যায়। যাক, আমরা এবার নতুন বোতামের কোড লিখবো। তবে এর আগে আরো দু'টো বোতাম ফর্মে যুক্ত করা জরুরী।

প্রথমটি হলো একটি 'ক্যানসেল' বোতাম। নতুন ডাটা লেখার বোতামে ক্লিক করার পরও ব্যবহারকারী যাতে ইচ্ছা পরিবর্তন করতে পারেন সে সুযোগ তো থাকতে হবে। দ্বিতীয়টি হলো 'রেকর্ড নতুন ডাটা' নামকরণে একটি বোতাম। নতুন রেকর্ড ফিল্ডে প্রবেশ করার পর এ বোতামে ক্লিক করে ব্যবহারকারী আমাদের ডাটাবেইজ 'ঠিকানাডিবি' -তে তথ্য যুক্ত করবেন- মূলত নতুন একটি তথ্য সারি যোগ হবে। আমরা বোতামগুলো যুক্ত করে একটি ইউই [User Iterface] তৈরি করেছি। উপরের চিত্রটি দেখুন। লক্ষ করুন, **'ক্যানসেল'** বোতামের (Name) Propery হবে: btnCancel। এর Visible Property হবে: False। **'রেকর্ড নতুন ডাটা'** বোতামের (Name) Propery হবে: btnRecord। এটার Enabled Property হবে: False। এবার আমরা কোডগুলোর প্রতি দৃষ্টি দিতে পারি।

নতুন রেকর্ড যুক্ত করার কোডটি কিছুটা জটিল। সুতরাং খুব সতর্কসহ ব্যাখ্যাটুকু পাঠ করে নেবেন। অবশ্য ইতোমধ্যে যেসব সাবরুটিন সম্পর্কে বলেছি এবং সেগুলো এখনও পরিবর্তিত আছে ওগুলোর ওপর ব্যাখ্যর পুনরাবৃত্তি করবো না। আমরা পুরো প্রোগ্রাম শেষে সমস্ত কোড আবার একসাথে উল্লেখ করবো- অবশ্যই। প্রথমেই যে কাজটি করা জরুরী তাহলো কিছু ডাইমেনশন [Dim] ডিক্লারেশন সর্বত্র প্রাপ্তিযোগ্য করতে হবে। এজন্য অবশ্য ফর্মের ক্লাস ডিক্লারেশনের পরই এসব কোড লিখতে হবে। অর্থাৎ গ্লোবাল ডিক্লারেশন হবে নিম্নরূপ:

```
'গ্লোবাল ডিক্লারেশন
    Dim RecNum As Integer = 0 'এটা হোল্ড করবে টেক্সট বাক্সের সংখ্যাটি যার দ্বারা সঠিক রো এর তথ্য আনা হবে।
    Dim NumRec As Integer 'এটা হোল্ড করবে টেবিলের মোট রেকর্ড নাম্বার।
    Dim KeepingIndex As Integer 'নতুন রেকর্ড তৈরিতে প্রদর্শিত রেকর্ডের ইনডেক্স নাম্বার সংরক্ষণ
    Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
    Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
    Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
    Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে।
    Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
    Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম হোল্ড করবে।
    Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করবে।
```

সবগুলো ভেরিয়েবলের ডিক্লারেন শেষে কমেন্ট লাইনে বুঝিয়ে দিয়েছি। সুতরাং এখানে আর কিছু যোগ করার নেই- শুধু একটি ছাড়া। এ ব্যাপারটি হলো ভেরিয়েবলগুলোর ডাটা টাইপ। মোট তিনটি ইন্টেজার [আস্ত সংখ্যা], চারটি স্ট্রিং [লেখা] এবং তিনটি অবজেক্ট টাইপ ডিক্লারেশন আছে। এগুলো কিভাবে কাজ করে তা পরবর্তিতে বুঝিয়ে বলছি।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

এবার প্রথম সাবরুটিনটি এখানে তুলে ধরছি:

```
' ডাটা আনুন বোমাতের ক্লিক ইভেন্ট। ব্যবহারকারী প্রথমেই এ বোতামে ক্লিক করবেন।
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    GetAndFillRecord() 'তথ্য আনার সাব কল।
    btnNewRecord.Enabled = True 'এই লাইটটি দ্বারা রেকর্ড নতুন ডাটা বোতামটি সক্রিয় করা হয়েছে।
End Sub
```

আমাদের ঠিকানা ডাটাবেইজ রান করলেই ঠিকানা টেবিল দেখতে চাইলে উপরোক্ত বোতামে ক্লিক করা ছাড়া বিকল্প কিছু নেই। সুতরাং এটাই প্রথমে একশনে যাবে। আমরা সুবিধার জন্য তথ্য এনে প্রদর্শনের কোড ব্লকটি আলাদা একটি সাবরুটিনে করেছি। প্রথম লাইন সেই সাবরুটিন কল করে ও সাথে সাথে নতুন রেকর্ড ঢুকানোর জন্য এর প্রোপার্টি সক্রিয় করে দেয়। ব্যস! এ বোতামের কাজ এটুকুই।

```
'রেকর্ড থেকে তথ্য এনে ফিল্ডে দেখানো হলো এ সাবের কাজ
Private Sub GetAndFillRecord()
    RecNum = Val(TextBox1.Text) 'ভেরিয়েবলে টেক্সবাক্সের সংখ্যা। প্রথমবার এর মান ০।
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
    FullDatabasePath = "C:\Users\htci\Desktop\LearningDotNet\MyDatabase\TikanaBoi.accdb"
    dbSource = "Data Source=" & FullDatabasePath 'dbSource এর মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
    If con.State <> ConnectionState.Open Then 'কানেকশন স্ট্রিং আছে কি না দেখতে হবে।
        con.ConnectionString = dbProvider & dbSource 'কানেকশন স্ট্রিং পূর্ণ হলো।
        con.Open()
    End If
    Dim cmd As New OleDb.OleDbCommand("Select COUNT(*) FROM Tikana", con)
    NumRec = cmd.ExecuteScalar() 'সিস্টেম ফাংশন দ্বারা রেকর্ড সংখ্যা আনা।
    'ফাংশন কল করে রেকর্ড সংখ্যা সঠিক আছে কি না দেখা।
    If ValidateEntryNum() = False Then
        con.Close() 'ভুল হলে ডাটাবেইজ বন্ধ করে এক্সিট করা
        Exit Sub
    End If
    Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
    'ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
    da = New OleDb.OleDbDataAdapter(Sql, con)
    da.Fill(ds, "TikanaDB") 'দ্বিতীয় প্যারামিটার যে কোনো কিছু হতে পারে।
    IDText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(0) 'আইডি
    NameText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(1) 'নাম
    AddressText1.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(2) 'ঠিকান-১
    AddressText2.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(3) 'ঠিকানা-২
    MobileText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(4) 'মোবাইল
    EnableNavigationBtn() 'ডাটাবেইজ লোড হয়েছে তাই চারটি ন্যাভিগেশন বোতাম সক্রিয় করতে হবে।
    TextBox2.Text = NumRec 'রেকর্ড সংখ্যা টেক্সট ফিল্ডে দেখানো।
End Sub
```

লক্ষ করুন, উপরোক্ত মূল সাবে বেশ রদবদল হয়েছে। প্রথমত, একটি মাত্র Dim স্টেটমেন্ট আছে। বাকিগুলো গ্লোবেল হিসাবে পূর্বেই ডিক্লায়ার করা হয়েছে। দ্বিতীয়ত একটি অতিরিক্ত IF--End If ব্লক যুক্ত হয়েছে। এছাড়া শেষের দিকের একটি লাইন এখানে নেই। সেটা হলো: con.Close()। এর কারণ হলো, এ পর্যায়ে আমরা ডাটাবেইজের সঙ্গে সংযুক্তি ক্লোজ করতে চাই না। পরিবর্তিত কোডগুলোর ব্যখ্যা পরের পৃষ্ঠায় এসেছে।

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

পরিবর্তিত কোড

মূল `GetAndFillRecord()` নামক সাবরুটিনের কোডে নিম্নলিখিত পরিবর্তন এসেছে:

১.
```
If con.State <> ConnectionState.Open Then 'কানেকশন স্ট্রিং আছে কি না দেখকে হবে।
    con.ConnectionString = dbProvider & dbSource 'কানেকশন স্ট্রিং পূর্ণ হলো।
    con.Open()
End If
```

এ ব্লকটির মাধ্যমে টেস্ট করা হয়, ডাটাবেইজের সঙ্গে কানেকশন খোলা আছে কি না। থাকলে কিন্তু পুনরায় খোলার নির্দেশ আসলে কম্পাইলারে অভিযোগ করবে। আমরা একাধিকবার এই সাবটি কোড রানিংয়ের সময় ডাকবো- তাই এ টেস্ট। বুঝতেই পারছেন, খোলা না থাকলে দ্বিতীয় লাইন কানেকশন স্ট্রিং প্রভাইড করবে ও তৃতীয় লাইন ডাটাবেইজের সঙ্গে কানেকশন সৃষ্টি করবে। এই সাবে এটুকুই পরিবর্তিত হয়েছে। শেষের লাইনটি যে কেটে দেওয়া হয়েছে তা-তো ইতোমধ্যে কারণসহ উল্লেখ করেছি।

২.
```
'new record button click event sub
Private Sub btnNewRecord_Click(sender As Object, e As EventArgs) Handles btnNewRecord.Click
    If isAllRecEntered() = False Then 'not a new record
        Beep()
        Exit Sub
    End If
    Dim RetVal As Integer = MsgBox("নতুন রেকর্ড লিখবেন?", MsgBoxStyle.YesNo)
    If RetVal = 7 Then Exit Sub
    KeepingIndex = Val(TextBox1.Text) 'keep the old index number incase cancelled
    IDText.Text = NumRec 'num of Record always 1 more then o index record number. No editing
    TextBox2.Text = NumRec + 1 'show number of record will gone up by one. No editing
    DeleteTextFromFields() 'call sub to delete fields texts
    DisableNavigationBtn() 'call sub to disable navigation buttons.
    btnRecord.Enabled = True
End Sub
```

এই সাবটি সম্পূর্ণ নতুন। এটা 'নতুন রেকর্ডের প্রস্তুতি নিন' বোতামের সাবরুটিন। এতে ক্লিক করলেই প্রথমে একটি টেস্ট করতে হবে। প্রথমেই দেখতে হবে কোনো ফিল্ডে টেক্সট আছে কি না। আমরা এখানে একটি ফাংশন কল করেছি। এটির কাজ হলো সকল টেক্সট ফিল্ডে লেখা থাকলে 'সত্য' এবং না থাকলে 'মিথ্যা' রিটার্ন করা। যদি 'মিথ্যা' রিটার্ন করে তাহলে বুঝা যাবে, ইতোমধ্যে আমরা নতুন রেকর্ডের জন্য নির্দেশ করে নিয়েছি। অন্যকথায় বোতামে ক্লিক করে শূন্য ফিল্ডগুলো নিয়ে এসেছি। পরের লাইনটি সতর্কবার্তা। নতুন রেকর্ড সত্যিই লিখা হবে কি না- তা পুনরায় জেনে নেওয়া। অবশ্য এখানেও ব্যবহারকারী ভুল করলে, পরবর্তিতে আরেকটি ক্যানসেল বোতাম আছে। সেটিতে ক্লিক করে মূল রেকর্ডে ফিরে যেতেও পারবেন। লক্ষ করুন, মেসেজ বাক্সে 'No' ক্লিক করলে রিটার্ন ভ্যালু হয় ৭। এজন্যই আমরা পরের লাইনে বলেছি RetVal ৭ হলে এক্সিট করার জন্য। জেনে রাখুন, ব্যবহারকারী ডায়লগে 'Yes' ক্লিক করলে রেজাল্ট হবে ৬।

এখন, মনে করুন ব্যবহারকারী ইয়েস চয়েজ করলেন। কিন্তু পরে আবার ক্যানসেলে ক্লিক করলেন, তখন তো আগে প্রদর্শিত তথ্যগুলো পুনরায় প্রদর্শন করতে হবে। তথ্যাদি প্রদর্শিত হয় টেবিলের [0-Based] ইনডেক্সের মাধ্যমে। আমাদের `TextBox1` -এ এই ইনডেক্স নাম্বারটি আছে। সুতরাং `NumRec` নামক আমাদের গ্লোবাল ভেরিয়েবলে বাক্সের সংখ্যাটি সংরক্ষণের ব্যবস্থা করা হয়েছে। নতুন রেকর্ড প্রবেশের সকল ফিল্ড খালি হবে, তাই এ সংখ্যাটি থাকলে ক্যানসেল বোতামে ক্লিক করলেই পুনরায় আগের রেকর্ডটি ডাটাবেইজ থেকে প্রদর্শন নিশ্চিত করা যাবে। পরের লাইন দ্বারা রেকর্ড সংখ্যটি [আইডি] ফিল্ডে প্রদর্শিত হয়েছে। টেক্সট বাক্স-২ এ প্রদর্শিত হয়েছে [নতুন রেকর্ডসহ] রেকর্ড সংখ্যা।

এবার আমরা কল করেছি `DeleteTextFromFields()` সাবরুটিন। এর কাজ হলো চারটি এনট্রি ফিল্ড মুছে দেওয়া। পরের লাইনও `DisableNavigationBtn()` নামক সাবরুটিন কল। এটার কাজ সকল ন্যাভিগেশন বোতাম অক্রিয় করা। সব শেষে নির্দেশ দিচ্ছি, 'রেকর্ড নতুন ডাটা' বোতামটি সক্রিয় করে দিতে- এরই মাধ্যমে আমরা নতুন রেকর্ড ঢুকাবো।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

৩.
```
'disables all four navigation btns
Public Sub DisableNavigationBtn()
    btnLast.Enabled = False : btnFirst.Enabled = False
    btnNext.Enabled = False : btnPrev.Enabled = False
    btnCancel.Visible = True
End Sub
```

এই সাবরুটিনের কাজ হলো চারটি ন্যাভিগেশন বোতাম অক্রিয় ও শুধুমাত্র ক্যানসেল বোতামটি প্রদর্শন করা। লক্ষ করুন, কলোন (:) এর ব্যবহার। একই লাইনে একাধিক নির্দেশ দিতে চাইলে এভাবে কলোন ব্যবহার করতে হয়। এই সাবরুটিনটি নতুন রেকর্ডের প্রস্তুতির অংশ।

৪.
```
'deletes five text fields and makes 4 of them editable
Public Sub DeleteTextFromFields()
    NameText.Text = "" : AddressText1.Text = ""
    AddressText2.Text = "" : MobileText.Text = ""
    NameText.ReadOnly = False : AddressText1.ReadOnly = False
    AddressText2.ReadOnly = False : MobileText.ReadOnly = False
    TextBox1.Text = "" 'we dont't want user to edit that field
    Button1.Enabled = False 'disable get record btn
End Sub
```

এটাও নতুন রেকর্ড প্রবেশ করার জন্য প্রস্তুতির অংশ। লক্ষ করুন, ডিজাইন টাইমে আমরা চারটি তথ্যফিল্ড অক্রিয় করে রেখেছিলাম। কারণ, আমরা চাইনি ব্যবহারকারী অসতর্কতাহেতু কোনো রেকর্ডে পরিবর্তন আনেন। তবে এবার তো নতুন রেকর্ড লিখা হবে তাই প্রথমে সবগুলে মুছে ফেলার পর এগুলো সক্রিয় করেছি। কিন্তু TextBox1 -এ কিছু লেখা আমরা চাই না। তাই একে মুছার পরও অক্রিয় রেখেছি। সব শেষের লাইন দ্বারা ‘ডাটা আনুন’ বোতামটি অক্রিয় করা হয়েছে।

৫.
```
'this sub locks all 4 flds
Public Sub MakeFourFldReadOnly()
    NameText.ReadOnly = True : AddressText1.ReadOnly = True
    AddressText2.ReadOnly = True : MobileText.ReadOnly = True
End Sub
```

রেকর্ড ঢুকানো কিংবা ক্যানসেল হওয়ার পর চারটি তথ্যফিল্ডকে পুনরায় অক্রিয় করতে হবে- যাতেকরে ব্যবহারকারী আর কিছু লিখতে না পারেন। এ কাজটিই আঞ্জাম দেয় উপরোক্ত সাবটি।

৬.
```
'retuns true if all 4 flds contains text, ready for record validation
Function isAllRecEntered()
    If NameText.Text <> "" And AddressText1.Text <> "" _
       And AddressText2.Text <> "" And MobileText.Text <> "" Then
        Return True
    End If
    Return False
End Function
```

ইতোমধ্যে এ ফাংশনটির কথা উল্লেখ করেছি। এর কাজ হলো টেক্স চারটিতে লেখা আছে কি না সে বুলিয়ান তথ্য রিটার্ন করা।

৭.
```
'btnCancel event code, simply show the last record, disable itsef, enable get rec btn
Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles btnCancel.Click
    TextBox1.Text = KeepingIndex 'এটার মধ্যে আগে প্রদর্শিত রেকর্ড ইনডেক্স আছে।
    GetAndFillRecord() 'রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
    MakeFourFldReadOnly() 'ফিল্ড চারটে রীড-অনলি বানানো।
    btnCancel.Visible = False : Button1.Enabled = True 'enable get record btn
    btnRecord.Enabled = False : btnNewRecord.Enabled = True
End Sub
```

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

আগের পৃষ্ঠার ৭ নাম্বারে প্রদর্শিত 'ক্যানসেল' বোতামের সাবরুটিন দ্বারা ইতোমধ্যে যে রেকর্ডটি টেক্স ফিল্ডগুলোতে ছিলো তা পুনরায় দৃশ্যমান করা হয়েছে। এটা প্রথমে ইনডেক্স সংখ্যা টেক্সবাক্স-১ এ লিখে দেয়। এরপর কল করে একে একে ইতোমধ্যে আলোচিত [GetAndFillRecord() এবং MakeFourFldReadOnly() নামক] দু'টি সাবরুটিন। প্রথমটি তো আমাদের মূল সাব- যার কাজ রেকর্ড প্রদর্শন করা আর দ্বিতীয়টির কাজ হলো চারটি ফিল্ডকে লেখার জন্য অক্রিয় করে দেওয়া।

৮.
```
    'record Button
Private Sub btnRecord_Click(sender As Object, e As EventArgs) Handles btnRecord.Click
    Dim MsgStr As Integer = MsgBox("সতর্কবার্তা! সবকিছু সঠিক মতো লিখেছেন তো?", MsgBoxStyle.YesNo)
    'MsgBox(MsgStr)
    If MsgStr = 7 Then
        Exit Sub
    End If
    AddRows()
End Sub
```

এই btnRecord ['রেকর্ড নতুন ডাটা'] বোতামের ক্লিক ইভেন্ট কল করে সাবরুটিন 'AddRows()'। এটা ডাকার পূর্বেও একটি সতর্কবার্তা ডায়লগ দেখানোর ব্যবস্থা করেছি। কারণ ব্যবহারকারী ভুল করতে পারেন তথ্য প্রবেশে- কিংবা তিনি নতুন রেকর্ড এ মুমূর্তে ঢুকানোর ক্রিয়া ক্যানসেলও করার ইচ্ছা রাখতে পারেন। তাকে এই সুযোগটুকু তো দিতে হবে না? প্রথম চার লাইন এ সুযোগ দিয়েছে।

৯.
```
Private Sub AddRows()
    If isAllRecEntered() = False Then
        MsgBox("সব রেকর্ড সঠিকভাবে ঢুকিয়েছেন কি না দেখুন! তথ্য রেকর্ড হয় নি।")
      Exit Sub
    End If
    EntryToRaw() 'রেকর্ড ঢুকানোর সাবরুটিন কল করা।
GetAndFillRecord() 'রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
MakeFourFldReadOnly() 'ফিল্ড চারটে রীড-অনলি বানানো।
    btnCancel.Visible = False
    Button1.Enabled = True 'enable get record btn
    btnRecord.Enabled = False
    btnNewRecord.Enabled = True
End Sub
```

এই সাবরুটিনইটিই হলো রেকর্ড টেবিলে যোগ করার পূর্ব প্রস্তুতির সর্বশেষ ধাপ। EntryToRaw() নামক সর্বশেষ সাবরুটিন কলের পূর্বে একটি টেস্ট করা হয়। সকল ফিল্ডে টেক্সট প্রবেশ করা হয়েছে কিনা তা জানতে হবে। যদি সকল বাক্সে টেক্সট না থাকে, তাহলে মেসেজ বাক্সের মাধ্যমে সতর্কবাণী প্রদর্শন করে বেরিয়ে যেতে হবে সাবরুটিন থেকে। এরপর [সবটি ঠিকঠাক থাকলে] প্রথমে EntryToRaw() সাবরুটিন কল করে নতুন রেকর্ড তৈরি করা হয়। এরপর রেকর্ড প্রদর্শনের মূল সাবরুটিন কল করে আগের প্রদর্শিত তথ্যাদি টেক্সট বাক্সগুলোতে পুনরায় দেখানো হচ্ছে। এর পরের সাবরুটিন কল করে সকল টেক্স বাক্স রিড-অনলি করতে হবে। সবশেষে তিনটি বোতামের ক্রিয়া পরিবর্তন করা জরুরী: ক্যানসেল বোতাম গোপন করা; রেকর্ড আনার বোতাম সক্রিয় করা; রেকর্ড করার বোতাম নিষ্ক্রিয় করা; এবং নতুন রেকর্ডের প্রস্তুতি-বোতাম আবার সক্রিয় করা।

এ হচ্ছে সর্বশেষ ডাটা এনট্রি সাবের পূর্ব পর্যন্ত সাবগুলোর ক্রিয়ার বর্ণনা। আমরা এবার পরের পৃষ্ঠায় এক্সেস টেবিলে তথ্যাদি রেকর্ড করার ক্রিয়াটির পূর্ণাঙ্গ ব্যাখ্যা তুলে ধরবো।

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
'Code for data entry into access table.
   Private Sub EntryToRaw()
       Dim cb As New OleDb.OleDbCommandBuilder(da) 'commandBuilder Object creation
       Dim dsNewRow As DataRow 'new data row creation variable
       'notice here, ID column as number will be added automatically [AutoNumber]
       dsNewRow = ds.Tables("TikanaDB").NewRow() 'adding a row into TikanaDB
       dsNewRow.Item(1) = NameText.Text 'name text into 2nd column cell
       dsNewRow.Item(2) = AddressText1.Text 'Address text into 3rd column cell
       dsNewRow.Item(3) = AddressText2.Text 'address text2 into 4th column cell
       dsNewRow.Item(4) = MobileText.Text 'mobile number [as text] into 5th column text
       ds.Tables("TikanaDB").Rows.Add(dsNewRow) 'actual row of data in dataset
       da.Update(ds, "TikanaDB") 'the row of data in ds goes to the created row
       con.Close() 'now close the connection
       MsgBox("ডাটাবেইজে নতুন রেকর্ড সফলভাবে প্রবেশ করা হয়েছে।") 'show success message
   End Sub
```

সাবের প্রথম লাইন: কমান্ড বিল্ডার সিস্টেম অবজেক্টের নতুন ইন্সট্যান্ট ডিক্লারেশন (আমাদের নামকরণ: cb)। এতে একটি প্যারামিটার আছে- তাহলো ডাটা এডাপটার [আগেই ডিক্লায়ার করা এই এডাপটারের আমাদের কর্তৃক নামকরণ: da)।

দ্বিতীয় লাইন: একটি ডাটারো টাইপ ডিক্লারেশন [টেবিলে নতুন তথ্যলাইন সৃষ্টির জন্য এটি, আমাদের নামকরণ: dsNewRow]

তৃতীয় লাইন: উক্ত ডাটাটাইপ ভেরিয়েবলে ঠিকানা ডাটাবেইজের টেবিলে একটি নতুন রো [বা তথ্যলাইন] সৃষ্টি করে dsNewRow -তে সংরক্ষণের নির্দেশ। লক্ষ করুন, ds.Tables নির্দেশনার একটি প্যারামিটার হলো আমাদের TikanaDB। এখানে ds হলো পূর্বেই সৃষ্ট আমাদের DataSet। সবশেষে .NewRow() দ্বারা পয়েন্ট করা হচ্ছে নতুন রো সৃষ্টির ফাংশনের দিকে। শিক্ষার্থীদের প্রতি অনুরোধ, এসব সিস্টেম নির্দেশনা কিভাবে কাজ করে, সে সম্পর্কে কোনো ভাবনার প্রয়োজন নেই। এতো গভীরে যাওয়া আমাদের কাজ নয়- বরং ওসব কাজ সফটওয়্যার বিজ্ঞানীদের। অর্থাৎ মাইক্রসফটের ইঞ্জিনিয়ারদের। আমরা তাদের আবিষ্কার ও উদ্ভাবনগুলো কাজে লাগাবো কিভাবে- সেটাই জানতে হবে।

চতুর্থ থেকে সপ্তম লাইন: এই লাইনগুলো দ্বারা ব্যবহারকারী কর্তৃক চারটি ফিল্ডে প্রবেশকৃত টেক্সট ইনডেক্স নির্দেশনার মাধ্যমে ডাটাসেটে সংরক্ষণ করেছি। লক্ষ করুন, আইডি সংখ্যা সংরক্ষণের প্রয়োজন নেই। এটা এক্সেস নিজেই বানাবে। কারণ আমরা এই সেলকে AutoNumber ডটাটাইপ হিসাবে তৈরি করেছি। এ পর্যন্ত সকল নির্দেশনা ডাটাসেট এর মধ্যে ds নামকরণে মেমোরিতে সংরক্ষণ করেছি। সবকিছু প্রস্তুত হয়ে গেছে। এবার সত্যিকার অর্থে টেবিলের মধ্যে একটি তথ্যলাইন যোগ করার নির্দেশ দিতে হবে।

অষ্টম লাইন: এই লাইনটি কিভাবে কাজ করে তা একটু বুঝার দরকার। মনে রাখবেন, প্রতিটি ডট [.] হলো পয়েন্টার ইঙ্গিত। প্রথমে ডাটাসেট [ds] পয়েন্ট করছে আমাদের ঠিকানা টেবিলের দিকে (.TikanaDB)। এরপর পয়েন্ট করছে রো বা তথ্যলাইনের দিকে [.Raw]। সবশেষে পয়েন্ট করছে নতুন রো সৃষ্টির দিকে [.Add(dsNewRow)]। এ লাইনের কোড রান হওয়ার পর আমাদের এক্সেস টেবিলে একটি নতুন তথ্যলাইন বাস্তবে যোগ হবে। এতে শুধুমাত্র আইডি ফিল্ডে একটি ইউনিক [আলাদা] আইডি সংখ্যা থাকবে। বাকী চারটি সেল এখনও শূন্য। পরের লাইনের নির্দেশ দ্বারা এগুলো পূরণ করা হবে।

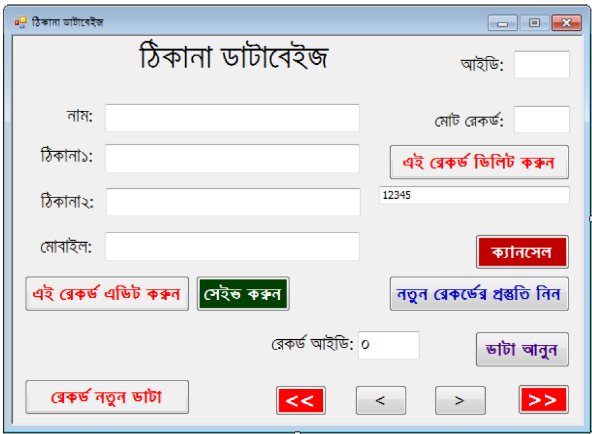
নবম লাইন: ds.Update একটি ফাংশন। এতে আছে দু'টি প্যারামিটার `(ds, "TikanaDB")`। প্রথমটি আমাদের সৃষ্ট ডেটাসেট যেখানে সংরক্ষিত আছে চারটি টেক্সট বাক্সের তথ্যাদি। দ্বিতীয় প্যারামিটার হলো আমাদের এক্সেস ডাটাবেইজের নাম। ব্যস! তথ্যাদি সেলের মধ্যে চলে যাবে।

দশম লাইন: কাজ শেষ। এবার কানেকশন বন্ধ করার পালা। এই লাইনে তা-ই নির্দেশ করা হয়েছে।

শেষ লাইন: ব্যবহারকারীকে তথ্য দেওয়া হচ্ছে- নতুন তথ্যলাইন তৈরি হয়েছে সফলভাবে। এখন প্রোগ্রাম বন্ধ করে পুনরায় চালালে নতুন রেকর্ডও দেখাবে।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

আমাদের ঠিকানা ডাটাবেইজের কাজ এখনো পুরোদমে শেষ হয় নি। যে কোনো তথ্য আপডেট করা ও মুছে দেওয়ার সুযোগ তো থাকতেই হবে। আমরা এখনই এ কাজগুলোর জন্য কোড লিখে নেবো। প্রথমে ফর্মের মধ্যে আরো তিনটি বোতাম যুক্ত করতে হবে। এর প্রথমটির টেক্সট্ প্রোপার্টি হবে 'এই রেকর্ড ডিলিট করুন'; নামকরণ: btnDelete। দ্বিতীয়টির টেক্সট্ প্রোপার্টি হবে 'এই রেকর্ড এডিট করুন'; নামকরণ: btnEdit। তৃতীয়টির টেক্সট্ প্রোপার্টি হবে 'সেইভ করুন'; নামকরণ: btnUpdate। সবগুলো বোতাম সুন্দরভাবে সাজিয়ে রাখুন। দেখুন বায়ের চিত্রটি। এবার কোডিং করতে হবে।



১. 'এই রেকর্ড ডিলিট করুন' বোতামের কোড: নতুন রেকর্ড করার কোড থেকে এ বোতামের কোড অনেকটা সহজ। দৃশ্যমান রেকর্ডটি ডিলিট হবে। বোতামের কোড প্রথমে দেখে নেই- এরপর ব্যাখ্যা আসবে।

```
'delete button's code
Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles btnDelete.Click
    ' first confirmation
    Dim ResP As Integer = MsgBox("আপনি কি সত্যিই এই রেকর্ডটি মুছে ফেলবেন?", MsgBoxStyle.YesNo)
    If ResP = 7 Then
        Exit Sub
    End If
    Dim RpT As String = InputBox("Enter Delete Record Password.")
        If ResP <> DeleteIdFld.Text Then
            MsgBox("Sorry, wrong Password! Record can not be deleted!", MsgBoxStyle.Critical)
            Exit Sub
        End If
    DeleteRow() 'call deleting sub
    If RecNum >= 0 Then
        GetAndFillRecord() 'রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
    End If
End Sub
```

প্রথমত, ব্যবহারকারীর জন্য একটি সেফটি নেটের প্রয়োজন। তিনি ভুলবশত, এ বোতামে ক্লিক করতে পারেন। সুতরাং একটি মেসেজ বাক্সে আবারো জিজ্ঞেস করা হচ্ছে, সত্যিই তিনি এই রেকর্ডটি মুছে দিতে চান কি না? অবশ্য আরো সতর্কতার জন্য একটি গোপন টেক্সটফিল্ডে নাম্বার বা পাস্ওয়ার্ড রাখা যায়। আমরা সেটাও করবো। ফর্মের মধ্যে একটি টেক্সট্ বাক্স নিয়ে আসুন, এর নামকরণ করুন: 'DeleteIdFld'। এর Visible Property হবে False। আইডি যা ইচ্ছা লিখতে পারেন। সহজ করতে যেয়ে আমরা বাক্সের Text Property -তে লিখবো: ১২৩৪৫। অবশ্য আমরা চাইলে সংখ্যাটি সরাসরি কোডে ঢুকিয়ে টেস্ট করতে পারতাম। তবে সুবিধার জন্য ফিল্ডে রাখাই ভালো। যাক উভয় কন্ডিশন পাস হওয়ার পরই আমরা **DeleteRow()** নামক সাবরুটিনটি কল করেছি। এটির কাজ হলো, যে রেকর্ড এ মুহূর্তে দৃশ্যমান আছে তা টেবিল থেকে মুছে ফেলা।

ডিলিট হওয়ার পর দেখতে হবে আর কোনো রেকর্ড আছে কিনা। এটা নিশ্চিত করতেই পরের If RecNum >= 0 ব্লকটি। যদি রেকর্ড থেকে থাকে, তখনই কল করেছি রেকর্ড পুনরায় ফিল্ডে নিয়ে আসার জন্য- আর তা হবে, এই মাত্র মুছে যাওয়া রেকর্ডের আগেরটি। এবার দেখে নিই রেকর্ড মুছে ফেলার সাবরুটিনটি। পরের পৃষ্ঠায় দেখুন।

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
'row delete codes
Sub DeleteRow()
    If NumRec = 0 Then 'incase there aren't any row of data
        MsgBox("টেবিলে তো আর কোনো রেকর্ড নেই!")
        Exit Sub
    End If
    Dim cb As New OleDb.OleDbCommandBuilder(da) 'commandBuilder Object creation
    ds.Tables("TikanaDB").Rows(RecNum).Delete() 'actual delete row command
    da.Update(ds, "TikanaDB") 'update database
    TextBox1.Text = RecNum - 1
    MsgBox("এই রেকর্ড মুছে গেছে।") 'show success message
    con.Close() 'close connection
End Sub
```

প্রথমে টেস্ট। যা কাটা হবে তা আছে কি? না থাকলে মেসেজ প্রদর্শন করে সাব থেকে বের হতে হবে। মনে রাখবেন, NumRec একটি গ্লোবাল ভেরিয়েবল, যার মধ্যে সংরক্ষিত আছে মোট রেকর্ড সংখ্যা- এটা ইনডেক্স সংখ্যা থেকে ১ বেশি। রেকর্ড থাকলে আমরা পরের লাইনে যেয়ে প্রথমে cb নামক একটি কমান্ড বিল্ডার অবজেক্ট ডিক্লায়ার করেছি। এর প্যারামিটার হলো ডাটাএডাপ্টার (da)। পরের লাইনে আমাদের ডাটাসেট [ds] এর মাধ্যমে পয়েন্ট করছি ঠিকানা ডাটাবেইজের টেবিল থেকে একটি তথ্যলাইন মুছে ফেলতে। Rows(RecNum) নির্দেশ করছে কোন্‌টি মুছা হবে। মনে রাখা দরকার, বর্তমানে প্রদর্শিত রেকর্ডের ইনডেক্স সংখ্যা হলো RecNum। সুতরাং আমরা এটিই ডিলিট করার কথা বলেছি। ঠিকানা ডাটাবেইজ এরপর আপডেট করতে হবে। বাস্তবে এই আপডেট নির্দেশই এক্সেসের ফিজিক্যাল ফাইলে রবদল নিশ্চিত করে। পরের লাইনের কাজ হলো ইনডেক্স সংখ্যা ১ কমানো- কারণ এখন একটি ইনডেক্স কমে গেছে। এরপর একটি মেসেজ বাক্সে রেকর্ড ডিলিটের বার্তাটি ব্যবহারকারীকে দেওয়া হয়েছে। আর সবশেষে কানেকশন বন্ধ করার নির্দেশ এসেছে।

এবার 'এই রেকর্ড এডিট করুন' ও 'সেইভ করুন' বোতামদ্বয়ের কোড নিয়ে আলোচনার প্রয়াস পাচ্ছি। আর এরই মাধ্যমে ডাটাবেইজ প্রোগ্রামিংয়ের ওপর আলোচিত এ পরিচ্ছেদের ইতি টানবো। শিক্ষার্থীরা এ থেকে যা শিখেছেন, সাধারণ ডাটাবেইজ প্রোগ্রামিংরে জন্য যথেষ্ট হবে বলে মনে করছি। তবে এর ওপর আরো গভীর জ্ঞানার্জনের জন্য ভিজুয়্যাল বেসিক.নেট দিয়ে ডাটাবেইজ সৃষ্টির উন্নতমানের বই-পুস্তক পাঠ ও ইন্টারনেটে প্রাপ্ত তথ্যাদি অধ্যয়ন করতে পারেন।

```
'editing btn code
Private Sub btnEdit_Click(sender As Object, e As EventArgs) Handles btnEdit.Click
    ' confimation test
    Dim ResP As Integer = MsgBox("এই রেকর্ডে রদবদল করবেন?", MsgBoxStyle.YesNo)
    If ResP = 7 Then
        Exit Sub
    End If
    EnableFourFlds() 'enable fields for editing
    btnUpdate.Visible = True 'show the update button
    Button1.Enabled = False
    btnDelete.Enabled = False
    btnNewRecord.Enabled = False
End Sub
```

আমাদেরকে আরোও একটি টেস্ট করতে হবে। কারণ, ব্যবহারকারী এই বোতামে ভুলেও ক্লিক করতে পারেন। টেস্টে পাশ হওয়ার পর প্রথমে ফিল্ডগুলো সক্রিয় করতে একটি সাবরুটিন ডাক দিয়েছি। এরপর সেইভ করুন বোতামটি দৃশ্যমান করেছি। এছাড়া 'নতুন রেকর্ডের প্রস্তুতি নিন', 'ডাটা আনুন' এবং 'এই রেকর্ড ডিলিট করুন' -এ তিনটি বোতামকে নিষ্ক্রিয় করতে হয়েছে। এগুলোর কোনো ফাংশন নেই এডিটের পূর্ব পর্যন্ত।

১৭তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles btnUpdate.Click
    Dim ResP As Integer = MsgBox("আপনি কি সব রেকর্ড সঠিকভাবে প্রবেশ করেছেন?", MsgBoxStyle.YesNo)
    If ResP = 7 Then
        Exit Sub
    End If
    UpdateRecord() 'call update sub
    MakeFourFldReadOnly() 'call sub to lock filds
    GetAndFillRecord() '□□□   □□"#□□□□$ □□%□□□□□ &□□ □□
    btnUpdate.Visible = False
    Button1.Enabled = True
    btnDelete.Enabled = True
    btnNewRecord.Enabled = True
End Sub
```

এখানেও একটি টেস্ট আছে! ব্যবহারকারী সঠিকভাবে সবকিছু এডিট করেছেন কি না জেনে নিতে হবে। টেস্টের ফলাফল ইতিবাচক হলে আমরা কল করতে পারি UpdateRecord() নামক সর্বশেষ সাবরুটিনটি। আপডেট হওয়ার পর চারটি ফিল্ড রিড-অনলি বানানো ও একটি বোতাম নিষ্ক্রিয় ও তিনটি সক্রিয় করা দরকার। শেষের একটি সাবরুটিন ও চারটি লাইন এ কাজগুলো সারবে।

এবার দেখে নেই সর্বশেষ কোড ব্লকটি। এটা একটি সাবরুটিন যার কাজ হলো এডিট করা লেখাগুলো স্ব স্ব স্থানে নিয়ে যাওয়া- তথা তথ্যাদি আপডেট করা।

```
Private Sub UpdateRecord()
    Dim cb As New OleDb.OleDbCommandBuilder(da) 'commandBuilder Object creation
    ds.Tables("TikanaDB").Rows(RecNum).Item(1) = NameText.Text
    ds.Tables("TikanaDB").Rows(RecNum).Item(2) = AddressText1.Text
    ds.Tables("TikanaDB").Rows(RecNum).Item(3) = AddressText2.Text
    ds.Tables("TikanaDB").Rows(RecNum).Item(4) = MobileText.Text
    da.Update(ds, "TikanaDB") 'update database
    MsgBox("রেকর্ড রদবদল হয়েছে।") 'show success
    con.Close() 'close connection
End Sub
```

সেইভ করুন [আপডেট করার] বোতামটির কোড বাস্তবে খুব এটা কঠিন কিছু না। প্রথমে -এর নতুন একটি ইন্সট্যান্ট OleDb.OleDbCommandBuilder সৃষ্টি করতে হবে। এর প্যারামিটার হবে আমাদের সৃষ্ট ডাটাএডাপটার [da]। এরপর চারটি লাইনে ডাটাসেটে নিয়ে যেতে হবে নির্দিষ্ট ইনডেক্সের [ds] চারটি আইটেম [১,২,৩ ও ৪]। সব প্রস্তুত। এবার da.Update(ds, "TikanaDB") লাইনটির মাধ্যমে এক্সেস ডাটাবেইজের টেবিলটি আপডেট করা হয়েছে। এরপর মেসেজ বাক্সের মাধ্যমে সফলতা জানানো ও কানেকশন ক্লোজ করা হয়েছে। আলহামদুলিল্লাহ! ঠিকানা ডাটাবেইজের কাজ পূর্ণাঙ্গভাবে শেষ হলো। শিক্ষার্থীদের প্রতি আবারো অনুরোধ, শুরু থেকে শেষ পর্যন্ত কোডের ব্যাখ্যা প্রয়োজনে একাধিকবার পাঠ করে বুঝার চেষ্টা করবেন। আর অবশ্যই, নিজে এই ডাটাবেইজ প্রোগ্রামটি সৃষ্টির প্রাকটিস করবেন।

শেষ হলো ডাটাবেইজ প্রেগ্রামিংয়ের ওপর রচিত এ পরিচ্ছেদটি। এবার আরো এক দু'টো স্টেপ বাই স্টেপ মডেল প্রোগ্রাম বানানোর প্রশিক্ষণ উপস্থাপনের চেষ্টা করবো। আসুন, আপনাদের সহযোগিতায় গড়ে তুলি ওসব ব্যবহারেপযোগী এ্যাপলিকেশন।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

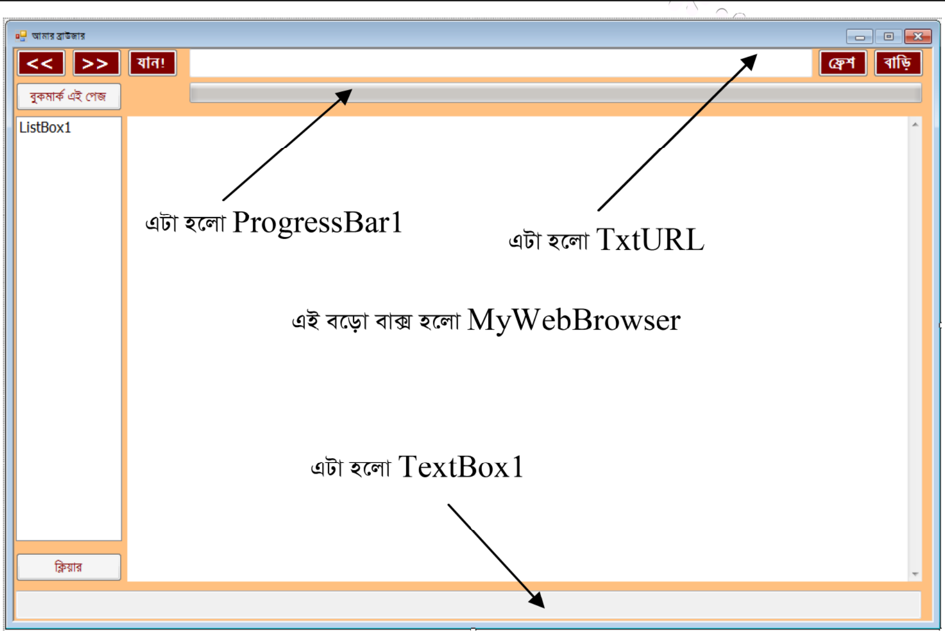
এ পরিচ্ছেদে আরো কয়েকটি নমুনা এ্যাপলিকেশন পদে পদে আমরা গড়ে তুলবো। শিক্ষার্থীরা প্রতিটি পদক্ষেপ সঠিকভাবে অনুসরণ করবেন, যেভাবে ইতোমধ্যে এ গ্রন্থে আলোচিত সকল প্রোগ্রাম তৈরির সময় করেছেন। পরিচ্ছেদ ১৭ শেষ হওয়ার পর ভেবেছিলাম ভিজুয়্যাল বেসিক.নেট এর উপর রচিত এই বইটির ইতি টানবো। কিন্তু শেষ পর্যন্ত সিদ্ধান্ত পরিবর্তন করলাম এ কারণে যে, কমিম্পউটার সফটওয়্যার সৃষ্টির মূলে প্রাকটিস একটি গুরুত্বপূর্ণ ব্যাপার। শিক্ষার্থীরা যতো বেশি প্রাকটিস করবেন ততো বেশি প্রোগ্রামিংয়ের বিভিন্ন কনসেপ্ট সম্পর্কে অভিজ্ঞতা অর্জন করবেন। হতে পারে, এ গ্রন্থের পাঠকদের মধ্যে ভবিষ্যতে দেশের একজন উজ্জ্বল তারকায়ও কেউ রূপান্তরিত হবেন। আমি অবশ্যই এ আশা করবো। যাক, আমরা একে একে এই নমুনা এ্যাপলিকেশনগুলোর ওপর বিস্তারিত আলোচনা শুরু করছি, এখনই।

১. একটি ব্যক্তিগত ওয়েভ ব্রাউজার: A Personal Web Browser

হ্যাঁ, ভিজুয়্যাল বেসিক.নেট ও ভিজুয়্যাল স্টুডিও আপনাকে ওয়েভ ব্রাউজার এ্যাপলিকেশনের মতো জটিল প্রোগ্রাম সৃষ্টির সুযোগ তৈরি করে দিয়েছে। আর তা এতোই সহজ! একটু পরই শিক্ষার্থীরা বুঝতে পারবেন। কয়েকটি বিশষ পদক্ষেপ ও অতি অল্প কিছু কোডিং দ্বারাই তৈরি হবে আপনার নিজস্ব স্বাপ্নিক ওয়েভ ব্রাউজার এ্যাপলিকেশন।

ক. ডিজাইন: নিচে একটি চিত্র দিয়েছি। আপনার ওয়েভ ব্রাউজার এরূপই হবে। এতে আছে ' টি বোতাম, ২টি টেক্সট বাক্স, ১টি লিস্ট বাক্স, ১টি প্রগ্রেস বার ও ১টি ওয়েভ ব্রাউজার কন্ট্রোল। এগুলোর অবস্থান এ্যারো দ্বারা ছবিতে দেখিয়েছি। এদের প্রত্যেকটির বিভিন্ন প্রোপার্টি নিচের টেবিলে তুলে ধরেছি। টুল বাক্স থেকে একে একে এগুলো ফর্মে নিয়ে আসুন। এরপর নির্দেশ মুতাবিক এদের প্রোপার্টি সেট করুন।

নাম	টেক্সট্ প্রোপার্টি	নাম	টেক্সট্ প্রোপার্টি	নাম	টেক্সট্ প্রোপার্টি
BtnBack	<<	BtnAddToFav	বুকমার্ক এই পেইজ	ListBox1	Font: Tahoma, 12pt
BtnForward	>>	BtnClear	ক্লিয়ার	ProgressBar1	--
BtnGo	যান!	TxtURL	--	MyWebBrowser	--
BtnRefresh	ফ্রেশ	TextBox1	ReadOnly: True	Form1	আমার ব্রাউজার
BtnHome	বাড়ি				



১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

২. কোডিং: আমরা ডিজাইন করে নিয়েছি। আপনি চাইলে ডিজাইনটি অন্যরকমও করতে পারেন। তবে বোতাম, টেক্সবাক্স, লিস্টবাক্স ইত্যাদি তো অবশ্যই থাকতে হবে। নিচে সম্পূর্ণ কোড তুলে ধরেছি। পরে ব্যাখ্যা আসবে।

```
Public Class Form1
  'যান! বোমাতের কোড। TxtURL -এতে পূর্ণাঙ্গ ওয়েভসাইট ঠিকানা লিখা থাকবে, তার ওপর অনুসন্ধান চালাবে এই বোতামটি।
   Private Sub BtnGO_Click(sender As Object, e As EventArgs) Handles BtnGo.Click
      MyWebBrowser.Navigate(TxtURL.Text) 'ন্যাভিগেট হলো সিস্টেমের ফাংশন। এর একটি প্যারামিটার হচ্ছে টেক্সট
      MyWebBrowser.Focus() 'ব্রাউজার বাক্সে ক্লিক করার অনুরূপ
   End Sub
  'এটা বাড়ি বোতামের কোড। আমরা গুগলস্ সার্চকে হোম পেইজ বানিয়েছি।
   Private Sub BtnHome_Click(sender As Object, e As EventArgs) Handles BtnHome.Click
      MyWebBrowser.Url = New System.Uri("https://www.google.com/webhp?hl=en&tab=wi")
      MyWebBrowser.Focus()
   End Sub
  'এটা আগের ওয়েভ পেইজে ফিরে যাওয়ার বোতাম।
   Private Sub BtnBack_Click(sender As Object, e As EventArgs) Handles BtnBack.Click
      MyWebBrowser.GoBack()
      MyWebBrowser.Focus()
   End Sub
  'এটা পরের ওয়েভ পেইজে যাওয়ার বোতাম।
   Private Sub BtnForward_Click(sender As Object, e As EventArgs) Handles BtnForward.Click
      MyWebBrowser.GoForward()
      MyWebBrowser.Focus()
      TextBox1.Text = MyWebBrowser.Url.ToString
   End Sub
  'এটা বর্তমান ওয়েভ পেইজ রেফ্রেশ করার বোতাম।
   Private Sub BtnRefresh_Click(sender As Object, e As EventArgs) Handles BtnRefresh.Click
      MyWebBrowser.Refresh()
      MyWebBrowser.Focus()
   End Sub
  'পেছনের ওয়েভ পেইজে যাওয়া শেষ হলে এই ইভেন্ট বোতামকে নিস্ক্রিয় করবে।
   Private Sub MyWebBrowser_CanGoBackChanged(sender As Object, e As EventArgs) Handles
MyWebBrowser.CanGoBackChanged
      BtnBack.Enabled = MyWebBrowser.CanGoBack
   End Sub
  'সামনের ওয়েভ পেইজে যাওয়া শেষ হলে এই ইভেন্ট বোতামকে নিস্ক্রিয় করবে।
   Private Sub MyWebBrowser_CanGoForwardChanged(sender As Object, e As EventArgs) Handles
MyWebBrowser.CanGoForwardChanged
      BtnForward.Enabled = MyWebBrowser.CanGoForward
   End Sub
  'ন্যাভিগেট হওয়ার পর এই ইভেন্ট সৃষ্টি হয়। এটে ধরে আমরা ইউআরএলকে উপরের ট্যাক্সট বাক্সে নিয়ে গেছি।
   Private Sub MyWebBrowser_Navigated(sender As Object, e As WebBrowserNavigatedEventArgs) Handles
MyWebBrowser.Navigated
      TxtURL.Text = MyWebBrowser.Url.ToString
   End Sub
  'ওয়েভ পেইজ লোড হওয়ার সময় প্রগ্রেস বারের ক্রিয়ার বোতাম।
   Private Sub MyWebBrowser_Navigating(sender As Object, e As WebBrowserNavigatingEventArgs) Handles
MyWebBrowser.Navigating
      With ProgressBar1 'উইথ--এন্ড উইথ ব্লক দ্বারা এটা বাড়ানোর রেইট নির্ধারণ করা হয়েছে।
         .Minimum = 0
         .Maximum = 52
```

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
            .Step = 4
        End With
            For index As Integer = 0 To 52 Step 4
                ProgressBar1.Value = index
                System.Threading.Thread.Sleep(35) 'নির্দিষ্ট সংখ্যক [৩৫] মিলিসেকেন্ডের জন্য বন্ধ থাকা
            Next
    End Sub
    'ফর্ম লোডের সময় ডিফোল্ট ওয়েব পেইজ খোলার নির্দেশ। আমরা গুগলসের সার্চ পেইজ নির্দিষ্ট করেছি।
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      MyWebBrowser.Navigate("https://www.google.com/webhp?hl=en&tab=wi")
        For Each item As String In My.Settings.Bookmarks
            ListBox1.Items.Add(item) 'বুকমার্কের সেটিংয়ে সংরক্ষিত ওয়েভপেইজ ঠিকানা লিস্ট বাক্সে নিয়ে আসা।
        Next
    End Sub
    'এটা বুকমার্ক ক্লিয়ার বোতামের কোড।
    Private Sub BtnClear_Click(sender As Object, e As EventArgs) Handles BtnClear.Click
        Dim ResP As Integer = MsgBox("সব বুকমার্ক মুছে দেবেন?", MsgBoxStyle.YesNo)
        If ResP = 7 Then 'no
            Exit Sub 'না মুছেই
        End If
        My.Settings.Bookmarks.Clear()
            ListBox1.Items.Clear()
        End Sub
    'এটা বর্তমান পেইজ বুকমার্ক বোতামের কোড।
    Private Sub BtnAddToFav_Click(sender As Object, e As EventArgs) Handles BtnAddToFav.Click
        My.Settings.Bookmarks.Add(MyWebBrowser.Url.ToString) 'সেটিংয়ে পেজ ঠিকানা যাবে।
        ListBox1.Items.Clear() 'লিস্ট ক্লিয়ার হবে।
     For Each item As String In My.Settings.Bookmarks
            ListBox1.Items.Add(item) 'সেটিংয়ের সকল পেজ ঠিকানা একে একে লিস্ট বাক্সে আসবে।
        Next
    End Sub
    'লিস্ট বাক্সে ডবল ক্লিক করে ওয়েভ পেইজে যাওয়ার কোড।
    Private Sub ListBox1_DoubleClick(sender As Object, e As EventArgs) Handles ListBox1.DoubleClick
        MyWebBrowser.Navigate(ListBox1.SelectedItem) 'লিস্ট বাক্সের আইটেম প্যারামিটার।
    End Sub
    'প্রাপ্ত ওয়েভ পেইজের ঠিকানা প্রাপ্তি ও এর ইউআরএল প্রদর্শন।
   Private Sub MyWebBrowser_StatusTextChanged(sender As Object, e As EventArgs) Handles
MyWebBrowser.StatusTextChanged
         TextBox1.Text = MyWebBrowser.StatusText 'প্রাপ্ত ওয়েভ পেইজের ঠিকানা নিচের টেক্সবাক্সে যাবে।
      End Sub
End Class
```

ব্রাউজারের মতো জটিল একটি এ্যাপলিকেশনের কোড এটুকুই মাত্র! আমরা মাইক্রসফটকে ধন্যবাদ না দিয়ে পারবো না। তারা .নেট ফ্রেমওয়ার্ক টেকনোলজি ডেভেলাপ করে প্রোগ্রামারদের জন্য বিরাট সম্ভাবনার দরোজা খুলে দিয়েছেন। ভিজুয়্যাল বেসিক.নেট ফ্রেমওয়ার্কের কিছু গুরুত্বপূর্ণ ক্লাস লাইব্রেবির যেগুলো টুলবাক্সের সঙ্গে সম্পৃক্ত, তাদের একটি লিস্ট গ্রন্থের পরিশিষ্টে উদাহরণ ব্যবহারসহ যুক্ত করার আশা করছি। যাক, এবার আমাদের ব্যক্তিগত ওয়েভ ব্রাউজারের কোডগুলোর ওপর প্রয়োজনীয় কিছু ব্যাখ্যা তুলে ধরছি।

প্রথমত জরুরী সবগুলো কোডের ব্যাখ্যা কমেন্ট দ্বারা করা হয়েছে। দ্বিতীয়ত কোড লাইনের অধিকাংশই স্ব-ব্যাখ্যাত-অর্থাৎ কোডই বুঝাচ্ছে তার ক্রিয়া কি হবে। তবে শিক্ষার্থীদের সুবিধার্তে ফর্মের ক্লাসের ভেতরস্থ ১৪টি সাবরুটিন সম্পর্কেই আরো কিছু আলোচনা করছি।

১. Private Sub BtnGO_Click -এটা হলো 'যান!' বোতামের ক্লিক সাবরুটিন। আপনি উপরের টেক্সবাক্সে যে ওয়েভ

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

ঠিকানা লিখবেন তা সঠিক হলে সেই ওয়েভ সাইট খুলবে।

২. Private Sub BtnHome_Click -এটা হলো 'বাড়ি' বোতামের ক্লিক সাবরুটিন। আমরা গুগল্স সার্চ পেইজকে বাড়ি বা হোম পেইজ বানিয়েছি।

৩. Private Sub BtnBack_Click -এটা হলো '<<' বোতামের ক্লিক সাবরুটিন। আগের পৃষ্ঠা [যদি থেকে থাকে], এই বোতাম আবার খুলবে।

৪. Private Sub BtnForward_Click -এটা হলো '>>' বোতামের ক্লিক সাবরুটিন। পরের পৃষ্ঠা [যদি থেকে থাকে], এই বোতাম আবার খুলবে।

৫. Private Sub BtnRefresh_Click -এটা হলো 'ফ্রেশ' বোতামের ক্লিক সাবরুটিন। বর্তমান পেইজটি পুনরায় লোড করবে এই ক্লিক ইভেন্টটি।

৬. Sub MyWebBrowser_CanGoBackChanged -এটা হলো 'GoBack' ফাংশনের একটি ইভেন্ট। এটা আত্মপ্রকাশ করে পেছনের পৃষ্ঠায় যেতে পারলে। আমরা এই ইভেন্টকে কাজে লাগিয়ে বলেছি আগের পৃষ্ঠায় যাওয়ার বোতাম সক্রিয় হতে পারে।

৭. Sub MyWebBrowser_CanGoForwardChanged -এটা হলো 'GoForward' ফাংশনের একটি ইভেন্ট। আমরা এ ইভেন্টকে কাজে লাগিয়ে বলেছি পরে যাওয়ার বোতামটি সক্রিয় হতে পারে।

৮. Sub MyWebBrowser_Navigated -এটা হলো 'Navigation' ফাংশনের একটি ইভেন্ট। এই ইভেন্ট সৃষ্টি হয় ওয়েভ পেইজ লোড হয়ে শেষ হওয়ার পরমুহূর্তে। আমরা একে ধরে সত্যিকার ইউআরএলকে স্ট্রিং টাইপে পরিবর্তন করে উপরের টেক্সট্ বাক্সে দেখিয়েছি।

৯. Sub MyWebBrowser_Navigating -এটাও ওয়েভ ব্রাউজারের একটি ইভেন্ট। পেইজ লোড হওয়ার সময় এটি সক্রিয় হয়। আমরা একে ধরে লোড হওয়ার দৃশ্যত কাজ প্রদর্শনের ব্যবস্থা করেছি প্রগ্রেসবার কন্ট্রোল ব্যবহার করে। লক্ষ করুন, ০ থেকে ১০০ পর্যন্ত ৪ করে অগ্রসর হওয়ার রেট নির্ধারিত করেছি প্রথমেই ইউথ-এন্ড উইথ ব্লকে। অর্থাৎ তার পুরোটাই ২৫টি পদক্ষেপে বাম দিক থেকে ডান দিকে যাবে। রেট নির্ধারিত হওয়ার পর একটি ফর -- নেক্সট্ লুপ দ্বারা সত্যিকার প্রগ্রেস হওয়ার গ্রাফিক্যাল এ্যানিমেশন দেখানো হয়েছে। আমরা .নেট এর এপিআই - System.Threading.Thread.Sleep(35) কাজে লাগিয়েছি। কতটুকু সময় প্রতিটি স্টেপে থামবে তা নির্ধারিত হয়েছে এই এপিআই থেকে। এখানে ৩৫ মিলিসেকেন্ড বুঝাচ্ছে [১০০০ মি.সে. = ১ সেকেন্ড]।

১০. Sub Form1_Load -এটা হলো ফর্ম লোড হওয়ার সময়কার ইভেন্ট। আমরা চাই প্রথমেই গুগল্স এর সার্চ ওয়েভ পেইজটি আসবে। সুতরাং ফর্ম খোলার পরই এই পেইজটিতে ন্যাভিগেট করার জন্য নির্দেশ দিয়েছি প্রথম লাইনে। এরপর আমরা এটাও চাই যে লিস্ট বাক্সে সেটিংয়ে সংরক্ষিত বুকমার্কসগুলো লাইনে লাইনে প্রদর্শন করতে। ফর ইচ্ -- নেক্সট্ লুপের কাজ এটি। শিক্ষার্থীদের একটি বিষয় জানা থাকা দরকার- কিভাবে সেটিংয়ে যেয়ে বুকমার্কস্ তৈরি করা যায়। এ ব্যাপারে পরের পৃষ্ঠার চিত্রগুলো দেখুন ও নিচের ব্যাখ্যাগুলো পাঠ করুন।

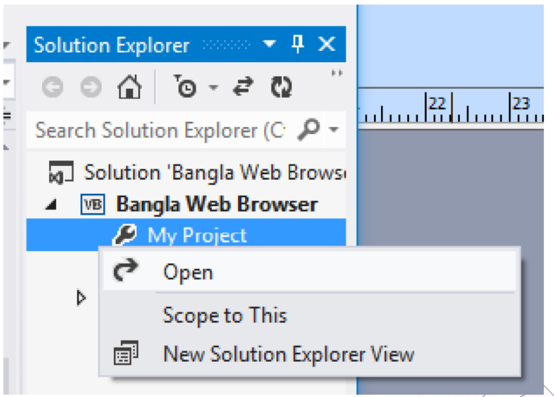
প্রথমে সলুশন এক্সপ্লোরারে যেয়ে মাই প্রজেক্টে রাইট-ক্লিক করুন। ফলে মাঝের জানালা আসবে। সেখানে সেটিং ক্লিক করুন। প্রথম কলামে উপরেই লিখুন Bookmark । এবার ডানের [...] বোতাকে ক্লিক করলেই একটি জানালা আসবে। সেখানে এই স্ট্রিংটি লিখুন: https://www.google.com/imghp?hl=en&tab=wi । এরপর OK ক্লিক করুন। ব্যস! সেটিংয়ে একটি স্ট্রিং সংরক্ষিত হবে। ভবিষ্যতে আপনি ব্রাউজারে প্রদর্শিত যে কোনো পেইজ সংরক্ষণের জন্য শুধুমাত্র **'বুকমার্ক এই পেইজ'** বোতামে ক্লিক করলেই সেটিংয়ের লিস্টে ঐ পেইজের ইউআরএল তালিকাভুক্ত হবে। আপনি যতোটা ইচ্ছা পেইজের ইউআরএল বুকমার্ক করতে পারেন। আর লিস্টের যে কোনো লাইনে ডবল ক্লিক করলেই সেই ইউআরএল-এ ন্যাভিগেট করবে ওয়েভ ব্রাউজার।

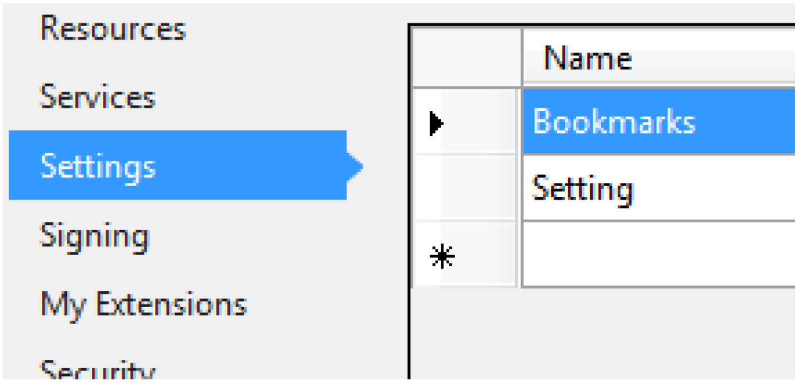
১১. Sub BtnClear_Click -এটা হলো 'ক্লিয়ার' বোতামের ক্লিক সাবরুটিন। বুকমার্ককৃত সকল পেইজের ইউআরএল এই বোতামে ক্লিক করে মুছে দিতে পারেন। অবশ্য একটি ওয়ার্নিং থাকবে- যা আমরা মেসেজ বাক্স ফাংশন দ্বারা প্রদর্শনের ব্যবস্থা করেছি।

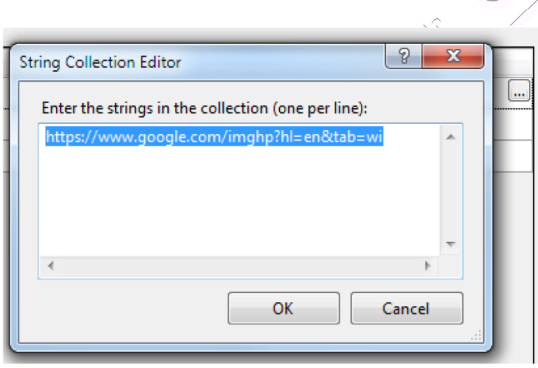
১২. Sub BtnAddToFav_Click -এটা হলো **'বুকমার্ক এই পেইজ'** বোতামের ক্লিক সাবরুটিন। প্রদর্শিত ইউআরএল সেটিংয়ের বুকমার্কস্ লিস্টে যাবে। তবে লিস্ট বাক্সেও নতুন বুকমার্ক আনতে হবে। এজন্য প্রথমে আমরা লিস্ট বাক্সের সকল আইটেম মুছে দিয়েছি দ্বিতীয় লাইনের নির্দেশ দ্বারা। এরপর একটি ফর ইচ-নেক্সট্ লুপ কাজে লাগিয়ে সেটিং থেকে পুনরায় সকল আইটেম নিয়ে

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*







এসেছি শূন্য লিস্টবাক্সে।

১৩. Sub ListBox1_DoubleClick -এটা হলো লিস্ট বাক্সের ডবলক্লিক ইভেন্ট। একে ধরে আমরা সিলেক্ট করা লাইন মুতাবিক ইউআরএল-এ ন্যাভিগেশন করার জন্য নির্দেশ করেছি ওয়েভব্রাউজারকে।

১৪. MyWebBrowser_StatusTextChanged -এটা হলো ওয়েভব্রাউজারের স্টেটাসটেক্সটচেঞ্জড নামক ইভেন্ট সাবরুটিন। আমরা একে ধরে প্রদর্শিত ওয়েভ পেইজের ইউআরএল লোড হওয়ার স্টেটাস টেক্সটটুকু নিচের টেক্সট বাক্সে প্রদর্নের জন্য নিয়ে এসেছি। লোড সম্পন্ন হয়ে গেলে, 'Done' শব্দটি বাক্সে দেখাবে। এবার দেখে নিন আপনার ব্রাউজারের একশন।



১৬তম পরিচ্ছেদ: *Few More Sample Applications*

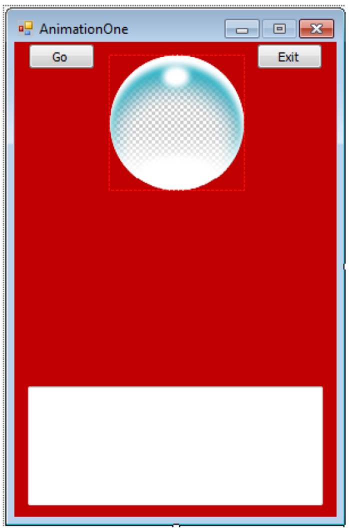
আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

২. সাধারণ এ্যানিমেশন: Simple Animation

এই গ্রন্থে আমরা বেশ কয়েকটি উদাহরণ প্রোগ্রাম নিয়ে বিস্তারিত আলেচনা করেছি। ভিজুয়্যাল বেসিক.নেট যে একটি শকিক্তশালী সফ্টওয়্যার ডেভেলাপমেন্ট ভাষা তা- তো নিশ্চয়ই পাঠকরা এখন অনুধাবন করে নিয়েছেন। এখন আমরা আরেকটি ছোট্ট উদাহরণ তুলে ধরছি। এটা হলো এ্যানিমেশন কোডিং উদাহরণ।

উদ্দেশ্য: ১. একটি বলকে উপর থেকে নিচে ধীরে ধীরে নামিয়ে আনবো। ২. একই সময় স্বয়ংক্রিয়ভাবে কিছু বাংলা লেখা আসবে একটি টেক্সট্ বাক্সে।

ক. ইন্টারফেইস ডিজাইন: একটি নতুন ইউন্ডোজ ফর্ম এ্যাপলিকেশন তৈরি করুন। এর নামকরণ যা ইচ্ছে করতে পারেন। আমি নামকরণ করেছি: Animation1। ফর্মের সাইজ হবে: ৩০০,৪৫৭। ফর্মের FormBorderStyle হবে - FixedSingle । ফর্মের মধ্যে থাকবে একটি টেক্সট্ বাক্স, দু'টি বোতাম ও একটি পিকচারবক্স। পিকচারবাক্সের BackgroundImage হবে যে কোনো একটি .png ছবি। ইন্টারনেটে প্রচুর ছবি মিলে। প্রয়োজনে ফটোশপে এনে পিএনজি হিসাবে বানাতেও পারেন। আমি একটি বল ব্যবহার করেছি। যেখানে ছবির ফাইল থাকবে সেখানে ব্রাউজ করে রিসোর্সে নিয়ে আসবেন। ছবির নাম যা থাকবে তা অবশ্য কোডে ব্যবহৃত হবে। ৩টি টাইমার কন্ট্রোল। এদের নামকরণ করুন: aTimer, wTimer, tTimer । টাইমারগুলো অবশ্যই ফর্মের উপরে থাকবে না। টাইমার কন্ট্রোলে ডবল ক্লিক করলেই এগুলো এসে যাবে। এরপর প্রতিটির প্রোপার্টি দেখে নামকরণ বদলাবেন। নিচের চিত্রে আমাদের ইন্টারফেইস ডিজাইনটি তুলে ধরেছি- একটু দেখে নিন।



খ. কোডিং: এ্যানিমেশন কোডিং কিছুটা জটিল- কিন্তু তেমন কঠিন নয়! প্রথমেই ফর্মে ডবল ক্লিক করে কোড জানালা খুলুন। সাবরুটিনগুলো শুরুর আগে দেখানো স্থানে ৫টি গ্লোবাল ডিক্লারেশন দরকার:

```
Public Class Form1
   Dim waitTime As Integer = 500 'millisecond to control speed of movement
   Dim speed As Integer = 60 'ms actual speed of movement
   Dim OXpos As Integer : Dim OYPos As Integer 'to keep record of the PicuteBox position in the form
   Dim speedText As Integer = 500 'ms to cotrol typing speed
   'text to be typed. Unicode is typed with Bijoy Bayanno Unicode option.
   Dim printText As String = "ভিজুয়্যাল বেসিক.নেট -নতুন শিক্ষার্থীদের জন্য গুরুত্বপূর্ণ প্রশিক্ষণ রিসোর্স - গ্রন্থকার ইঞ্জিনিয়ার আজিজুল বারী, ডিপ্লোমা ইন সাইন্স এন্ড টেকনোলজি, ইংল্যান্ড।"
```

এগুলো আমরা পরে ব্যবহার করবো। এরপর নিচের সাবরুটিনটি যাবে। এটা হলো aTimer -এর Tick নামক ইভেন্ট সাবরুটিন।

```
'this sub animates picture and types text
Private Sub aTimer_Tick(sender As Object, e As EventArgs) Handles aTimer.Tick
    TextBox1.Text += printText.Substring(TextBox1.Text.Length, 1) 'one by one printText writing will be printed in the text box
    If TextBox1.Text = printText Then aTimer.Enabled = False 'Once writing is finished, the timer action will stop
    'same time picture box's location will change by y+1 (1 point down). Notice x value doesn't change.
    MyPictureBox.Location = New Point(MyPictureBox.Location.X, MyPictureBox.Location.Y + 1)
    If MyPictureBox.Location.Y = 205 Then '250 point (y) is the minimum.
      aTimer.Enabled = False 'at that point timer stops
    End If
End Sub
```

উপরের সাবটিই মূল এ্যানিমেশনের কাজটি করে। পরের সাবরুটিন দু'টি টাইমার কন্ট্রোল সক্রিয় ও একটি অক্রিয় করে।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
'this sub enales aTimer and tTimer and disables wTimer
   Private Sub wTimer_Tick(sender As Object, e As EventArgs) Handles wTimer.Tick
      wTimer.Enabled = False
      aTimer.Enabled = True
      tTimer.Enabled = True
   End Sub
```

এর পরেরটি হলো ফর্ম লোড ইভেন্ট সাবরুটিন। আমরা একে ধরে প্রথমে ডিক্লায়ার করা ভেরিয়েবলগুলোতে ভ্যালু প্রবেশ করেছি।

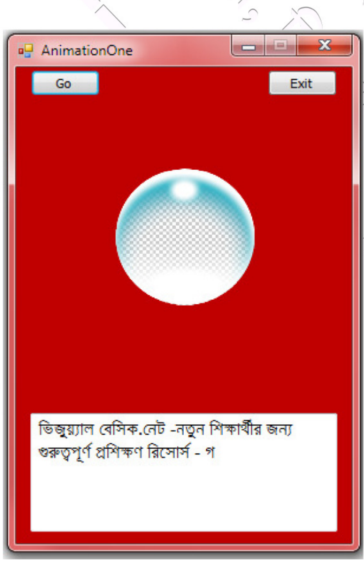
```
'form load-time inirialization of 3 variables associated with three timers
   Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      aTimer.Interval = speed
      wTimer.Interval = waitTime
      tTimer.Interval = speedText
      ' keep a record of PictureBox Position
      OXpos = MyPictureBox.Location.X : OYPos = MyPictureBox.Location.Y
   End Sub
```

নিচের ছোট্ট সাবটি হলো Exit বোতামের ক্লিক ইভেন্ট কোড। বুঝতেই পারছেন, এটি দ্বারা প্রোগ্রাম বন্ধ করা হয়।

```
 'exit- here Me means the Form itself.
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      Me.Close()
   End Sub
```

ব্যস! কোড মাত্র এটুকুই। এবার টেস্ট রানের পালা। নিচের তিনটি চিত্রে এ্যানিমেশনকালীন স্ক্রিনশট তুলে ধরেছি।







৩. বাংলা-ইংরেজি-বাংলা তারিখ রূপান্তর: Bangla-English-Bangla Date Conversion

ইনটারনেটে একাধিক ওয়েভসাইট আছে যেখানে বাংলা থেকে ইংরেজি এবং ইংরেজি থেকে বাংলা তারিখ রূপান্তরের সুযোগ আছে। কিন্তু কিভাবে এরূপ এ্যাপলিকেশন ডেভেলাপ করা যায়? এ প্রশ্নের জবাব খুঁজতে যেয়েই শেষ পর্যন্ত নিজেই নতুন এই প্রোগ্রাম সৃষ্টি করতে সক্ষম হয়েছি। আর এটি শেয়ার করতে চাই সকল শিক্ষার্থীদের সাথে। এই প্রোগ্রাম ডেভেলাপমেন্ট টেকনিক থেকে ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং ভাষা সম্পর্কেও অনেক ব্যাপার জানা ও সেসাথে অভিজ্ঞতা অর্জিত হবে। তাহলে আসুন, দেখে নিই কিভাবে এ ব্যবহারযোগ্য কার্যকরী এ্যাপলিকেশনটি বানাতে হয়।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

তথ্যাদি সংগ্রহ: প্রথমেই জানতে হবে বাংলা ও ইংরেজি তারিখের মধ্যে সম্পর্ক কি। কিছু অনুসন্ধানের পর জানা গেলো, বাংলা সন বের করতে ৫৯৩ বিয়োগ দিতে হবে ইংরেজি সন থেকে। তবে তারিখ যদি ১লা জানুয়ারি থেকে ১৩ এপ্রিলের মধ্যে হয় তাহলে কিন্তু বিয়োগ দিতে হবে ৫৯৪। এর কারণ হলো তখনো বাংলা সন বদলাবে না। এর পরের সকল তারিখের ক্ষেত্রে অবশ্য বিয়োগ দিতে হবে ৫৯৩। তাহলে সন পরিবর্তনের ফর্মুলা হলো:

১লা জানুয়ারি থেকে ১৩ এপ্রিল পর্যন্ত তারিখের ভেতর: ইংরেজি সন - ৫৯৪ = বাংলা সন। [বাংলা সন = +৫৯৪]
১৪ এপ্রিল থেকে ৩১ ডিসেম্বর পর্যন্ত তারিখের ভেতর: ইংরেজি সন - ৫৯৩ = বাংলা সন। [বাংলা সন = +৫৯৩]

তারিখ পরিবর্তের ক্ষেত্রে একটি ফর্মুলা জানা গেছে। তাহলো: **১৪ ১৫১৫ ১৬১৬১৬১৬ ১৫১৫ ১৪ ১৩ ১৫**। মনে রাখার জন্য আমরা লিখতে **৪ ৫৫ ৬৬৬৬ ৫৫ ৪৩৫**। এখানে ১লা বৈশাখ শুরু হয় ইংরেজি এপ্রিল মাসের ১৪ তারিখ। ১লা জৈষ্ঠ্য হলো ইংরেজি মে মাসের ১৫, ১লা আষাঢ় শুরু হয় ইংরেজি জুন মাসের ১৫ ইত্যাদি বুঝাচ্ছে। এই ফর্মুলা থেকে আমরা একটি তথ্যের টেবিল তৈরি করতে পারি। যেমন:

Month (1-12)	Day Range	Bangla M	Formula	month	Day Range	E.M	
4	14-30	1	-13	1	1-17	4	+13
5	1-14	1	+17	1	18-31	5	-17
5	15-31	2	-14	2	1-17	5	+14
6	1-14	2	+17	2	18-31	6	-17
6	15-30	3	-14	3	1-16	6	+14
7	1-15	3	+16	3	17-31	7	-16
7	16-31	4	-15	4	1-16	7	+15
8	1-15	4	+16	4	17-31	8	-16
8	16-31	5	-15	5	1-16	8	+15
9	1-15	5	+16	5	17-31	9	-16
9	16-30	6	-15	6	1-15	9	+15
10	1-15	6	+15	6	16-30	10	-15
10	16-31	7	-15	7	1-16	10	+15
11	1-14	7	+16	7	17-30	11	-16
11	15-30	8	-14	8	1-16	11	+14
12	1-14	8	+16	8	17-30	12	-16
12	15-31	9	-14	9	1-17	12	+14
1	1-13	9	+17	9	18-30	1	-17
1	14-31	10	-13	10	1-18	1	+13
2	1-12	10	+18	10	19-30	2	-18
2	13-28(29)	11	-12	11	1-16(17)	2	+12
3	1-14	11	+16	11	17(18)-30(31)	3	-16
3	15-31	12	-14	12	1-17	3	+14
4	1-13	12	+17	12	18-30	4	-17

লক্ষ করুন: ইংরেজি মাস কোন্‌টি সেটা জানতে পারলে আমরা কোন্ দু'টি তারিখের ভেতর কতো যোগ বা বিয়োগ করলে বাংল তারিখটি পাবো তা দেখানো হয়ছে। একই সময় বাংলা মাসটি কি হবে তাও উল্লেখিত আছে। অনুরূপ বাংলার ক্ষেত্রেও আমরা জানতে পারি কোন তারিখে কতোদিন যোগ বা বিয়োগ করতে হবে। একটি উদাহরণ থেকে

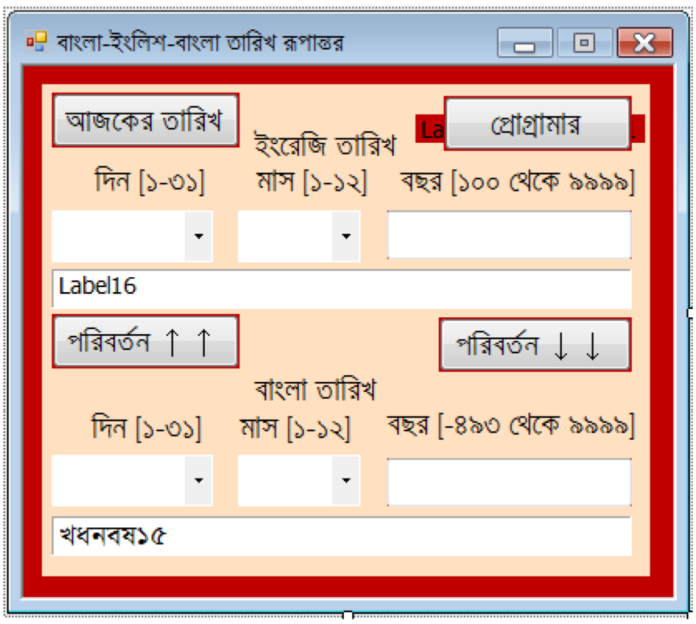
১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

বিষয়টি স্পষ্ট হবে। মনে করুন এই তারিখ: ২৫/৫/১৮৯৯ [জাতীয় কবি কাজী নজরুল ইসলামের জন্ম তারিখ] বাংলা তারিখে রূপান্তর করবো। আমরা প্রথমেই দেখবো কোন্ সংখ্যাটি বিয়োগ করে বাংলা সন বের করতে হবে। যেহেতু ইংরেজি মাসটি হলো মে [৫] যা ১৫ এপ্রিল থেকে ৩১ ডিসেম্বরের ভেতর পড়ে তাই আমাদেরকে বিয়োগ করতে হবে ৫৯৩ সংখ্যাটি। সুতরাং যে বাংলা সনে কবির জন্ম হয়েছিল তা হলো: ১৮৯৯ - ৫৯৩ = ১৩০৬ বাংলা।
এবার টেবিলের দিকে তাকিয়ে দেখতে পাচ্ছি যে, তারিখের দিনটি ইংরেজি মে (৫ম) মাসের ১৫-৩১ দিনের ভেতর হলে বাংলা তারিখের দিনটি হবে: ইংরেজি তারিখ - ১৪ = ২৫-১৪ = ১১। আর মাস হবে: দ্বিতীয়টি তথা জৈষ্ঠ্য। সুতরাং আমাদের জাতীয় কবির জন্ম হয়েছিল: ১১ জৈষ্ঠ্য, ১৩০৬ বাংলা। সুতরাং যে কোনো ইংরেজি তারিখ বাংলায় রূপান্তরের একটি সূত্র পেয়ে গেলাম। অনুরূপ বাংলা থেকেও ইংরেজিতে রূপান্তর করতে টেবিলের তথ্যাদি যথেষ্ট হবে। যেমন: আজকের [এই লেখা রচনার] তারিখ হলো ২৯ পৌষ, ১৪২৩ বাংলা। আমরা এটাও জানি, আজকের ইংরেজি তারিখ হলো ১২ জানুয়ারী, ২০১৭ ইং। আমরা টেবিল ব্যবহার করে ইংরেজি তারিখ বের করতে পারি এভাবে:
পৌষ মানে বছরের ৯তম মাস। আর তারিখটি হলো ২৯। সুতরাং টেবিলে যেয়ে দেখতে পাচ্ছি যে, ৯ম মাসের ১৮-৩০ দিনের ভেতর হলে বছর পড়ে জানুয়ারী। যেতেতু ১লা জানুয়ারী থেকে ১৩ এপ্রিল পর্যন্ত রেঞ্জের ভেতর তারিখ হলে যোগ দিতে হয় ৫৯৪ তাই ১৪২৩ বংলা = ১৪২৩ + ৫৯৪ = ২০১৭ ইং। তারিখ বের করতেও আমরা টেবিল ব্যবহার করতে পারি। বাংলা মাস ৯ (পৌষ) ও দিন ১৮-৩০ এর ভেতর হওয়ায় বিয়োগ হবে ১৭। সুতরাং ইংরেজি দিন হবে: ২৯ - ১৭ = ১২। তাই এখন বলতে পারি: ২৯ পৌষ, ১৪২৩ বাংলা = ১২ জানুয়ারী, ২০১৭ ইং।

সুতরাং আমরা যেভাবে উপরের তারিখগুলো পরিবর্তন করেছি তা-ই করতে হবে কম্পিউটারের সফ্‌টওয়্যারের মাধ্যমে।

ডিজাইন: ইউজার ইন্টারফেইস ডিজাইন একটি গুরুত্বপূর্ণ কাজ। আপনি চাইবেন ব্যবহারকারী যাতে সহজে এ্যাপলিকেশনটি কাজে লাগাতে পারেন। মোটকথা প্রয়োজনীয় সুযোগ-সুবিধাসহ একটি ব্যবহারকারী-বান্ধব ইউজার ইন্টারফেইস হলো সফল এ্যাপলিকেশন সৃষ্টির চাবিকাঠি। প্রথমে নিচের ডিজাইনটি দেখুন। এরপর কি কি বস্তু টুলবাক্স থেকে আনবেন ও এদের প্রোপার্টি কি হবে তা একটি টেবিলে তুলে ধরবো।



বস্তু	নাম	টেক্সট	ফন্ট-সাইজ
ফর্ম	.....	যে কোনো	.....
৪টি বোতাম	.....	যেভাবে আছে	যেভাবে আছে
১৬টি লেবেল	.....	যেভাবে আছে	শেষের ৬টির ভিজিবল= False
৪টি কম্বোবাক্স	....	....	বাংলা- ১৪
দু'টি টেক্সট বাক্স	....	....	বাংলা - ১৪

লক্ষ করুন: উপেরর বায়ের কম্বোর নাম: cmbEnglishDay। ডানেরটির নাম: cmbEnglishMonth। নিচের বায়েরটির নাম: cmbBanglaDay। ডানেরটি: cmbBanglaMonth। উপরের টেক্সট বাক্সের নাম: txtEnglishYear। নিচেরটির নাম: txtBanglaYear। এই নামগুলো আমরা কোডে ব্যবহার করেছি। পরিবর্তন বোতামের এ্যারোগুলো আনতে সিম্বল কাজে

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

লাগান। এবার দেখে নিই কোডগুলো।

কোড: আমরা কোডের বিভিন্ন সাবরুটিন ও ফাংশন ব্যাখ্যাসহ আলাদাভাবে তুলে ধরবো। এভাবে হলে কেডগুলো বুঝা অনেকটা সহজ হবে। কারণ, তারিখ পরিবর্তন কোড বাস্তবে তেমন সহজ নয়- বরং বেশ জটিল।

ফর্মের মধ্যে ডবল ক্লিক করে কোড এডিটর খুলুন। ফর্মের ক্লাস ডিক্লারেশন লাইন: Public Class Form1 -এর নিচে লিখুন এই গ্লোবাল ভেরিয়েবল ডিক্লারেশনগুলো:

```
Dim ToBanglaAdderVal As Integer 'স্টোর করে ইংরেজি দিনের সাথে কোন্ সংখ্যা যোগ হবে
Dim ToBanglaMonth As Integer 'স্টোর করে বাংলা মাস কি হবে
Dim ToBanglaYear As Integer 'স্টোর করে বাংলা বছর মান কি হবে
Dim IsitLeapYear As Boolean 'বুলিয়ান স্টোরস লিপ ইয়ার কি না - সত্য-মিথ্যা
Dim ToEnglishAdderVal As Integer 'স্টোর করে ইংরেজি দিনের সাথে কোন্ সংখ্যা যোগ হবে
Dim ToEnglishMonth As Integer 'স্টোর করে ইংরেজি মাস কি হবে
Dim ToEnglishYear As Integer 'স্টোর করে ইংরেজি বছর কি হবে
```

উপরোক্ত ভেরিয়েবলের কোন্‌টির কাজ কি তা কমেন্ট লাইনে লিখেছি। এবার প্রথম সাবরুটিনটি টাইপ করুন।

```
'প্রোগ্রাম খুললেই যে তারিখ বাংলা ও ইংরেজিতে হবে
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    PutInEnglishDate() 'সাব কল, এটা ইংরেজি আজকের তারিখ বসাবে।
    convertToBanglaDate() 'এ সাব ইংরেজি আজকের তারিখকে বাংলা করে বসাবে।
End Sub
```

আমরা এখানে দু'টি সাবরুটিন কল করেছি যেগুলো পরে ডিফাইন করা হয়েছে। এদের কাজ হলো বাংলা রূপান্তরসহ 'আজকের তারিখ' প্রদর্শন করা। সুতরাং ব্যবহারকারী প্রোগ্রাম রান করার পরই বর্তমান তারিখ ইংরেজি ও বাংলায় দেখতে পাবেন। সাবরুটিন **PutInEnglishDate()** -এর মাধ্যমে উপরের প্রথম কম্বোতে ইংরেজি দিন, পরের কম্বোতে মাস ও টেক্সট্ বাক্সে বছরের সংখ্যাটি দেখাবে। অনুরূপ নিচের প্রথম কম্বোতে বাংলা দিন, পরের কম্বোতে মাস ও টেক্সট্ বাক্সে বছরের সংখ্যা প্রদর্শন হবে দ্বিতীয় সাবরুটিন **convertToBanglaDate()** -এর মাধ্যমে। এগুলো কিভাবে কাজ করে তার ওপর আলোচনা সময়মতো হবে। পরের সাবটি হলো ইংরেজি থেকে বাংলায় পরিবর্তন বোতামের ক্লিক ইভেন্ট কোড।

```
'এটা ইংরেজি থেকে বাংলায় পরিবর্তনের বোতাম।
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    If Val(txtEnglishYear.Text) < 100 Or Val(txtEnglishYear.Text) > 9999 Then
        MsgBox("বছর রেঞ্জ হবে ১০০ থেকে ৯৯৯৯।")
        txtEnglishYear.Text = ""
        txtEnglishYear.Focus()
        Exit Sub
    End If
    Label15.Text = ""
    'লিপ ইয়ার না হলে ফেব্রুয়ারি ২৯ এর নিচে থাকতে হবে।
     If IsItLeap(Val(txtEnglishYear.Text)) = False Then
        If (cmbEnglishDay.SelectedItem) >= 29 And (cmbEnglishMonth.SelectedItem) = 2 Then
            MsgBox("Not a leap year! February month has 28 days only.", MsgBoxStyle.Exclamation)
            Exit Sub
        End If
    Else
        'যদি লিপ ইয়ার হয়ে থাকে তাহলে ফেব্রুয়ারি অবশ্যই ২৯ কিংবা কম থাকতে হবে।
          If (cmbEnglishDay.SelectedItem) >= 30 And (cmbEnglishMonth.SelectedItem) = 2 Then
              MsgBox("Its a leap year! February month has 28 days only.", MsgBoxStyle.Exclamation)
          Exit Sub
```

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
        End If
    End If
     'লিপ ইয়ার নয় বা লিপ ইয়ার হয়, এর পরও দিনের তারিখ সঠিক থাকলে
   UpdateLabel() 'গোপন লেবেলে দিন, মাস ও বছরের মান নিয়ে যেতে হবে, এই সাবের কাজ এ  টি।
  'এরপর দেখতে হবে দিনের মাস অন্যান্য মাসের বেলায়ও সঠিক আছে কি না।
    If IsDayEntryCorrect((Val(Label10.Text)), Val(Label9.Text)) = False Then
        MsgBox("উপস্! মাসে য দিন আছে তারচেয়ে বেশি হয়ে গেছে, বোধ হয়!", MsgBoxStyle.Exclamation)
        Exit Sub
    End If
     'এবার সব সঠিক হয়েছে
     convertToBanglaDate() 'কল কনভার্ট টু বাংলা সাবরুটিন।
End Sub
```

কোডের ব্যাখ্যা: প্রথম ইফ-এন্ড ইফ ব্লকের কাজ হলো বছরের সংখ্যাটি রেঞ্জের ভেতর আছে কি না তা নিশ্চিত করা। আমরা ইংরেজি ১০০ থেকে ৯৯৯৯ সাল পর্যন্ত রেঞ্জ দিয়েছি। এর বাইরে হলে প্রোগ্রাম মেসেজ দিয়ে কোড থেকে বের হবে। এরপর আমরা লেবেল ১৫ এর টেক্সট মুছে দিয়েছি। পরের ইফ-এন্ড ইফ ব্লকে আরেকটি টেস্ট করা হয়েছে। আমরা জানতে চাই, বছর সংখ্যাটি লিপ ইয়ার কি না। এ চেকটি করেছি একটি ফাংশন দ্বারা, যার নাম IsItLeap। এর একটিমাত্র প্যারামিটার আছে যা হলো বছর সংখ্যাটি। ফাংশন বুলিয়ান [হ্যাঁ, না] রিটার্ন করে। সুতরাং দেখতে হবে রিটার্ন করার পরও ব্যবহারকারী ফেব্রুয়ারি মাসের জন্য ২৮ দিন বা তার নিচে প্রবেশ করেছেন কি না। পরের ইফ-এন্ড ইফ ব্লকের কাজ এটি। কিন্তু লিপ ইয়ার হয়ে থাকলে দিন সংখ্যা যাতে ২৯ এর বেশি না হয় তা-ও দেখতে হবে। সুতরাং একই ইফ-এন্ড ইফ ব্লকেই Else কিওয়ার্ড দ্বারা আরো একটি চেক করতে হয়েছে। উভয় চেক সফল হওয়ার পরই পরবর্তী কোড এক্সেকিউট হতে পারে। প্রথমে আমরা একটি সাবরুটিন [`UpdateLabel()`] কল করে তারিখের সংখ্যাগুলো লুকানো তিনটি লেবেলে নিয়ে গেছি। এরপর আরো একটি সংখ্যা পরীক্ষা করা দরকার। মাসের মধ্যে য'দিন আছে তারচে বেশি প্রবেশ হয়েছে কি না তা-ও জানতে হবে। এই টেস্ট সফল হওয়ার পর আমরা অবশেষে প্রস্তুত হলাম পরিবর্তন সাবরুটিন কল করতে। এই সাবরুটিনের নাম `convertToBanglaDate()`। সঠিক সময়ে এটার ব্যাখ্যা আসবে।

ইতোমধ্যে বলেছি, আগের সাবরুটিনে `UpdateLabel()` নামক একটি সাবরুটিন কল করেছি। সুতরাং এটাই আমাদের পরবর্তী সাবরুটিন। এর কাজ হলো কম্বো বাক্স থেকে ইংরেজি দিনের সংখ্যা, ইংরেজি মাসের সংখ্যা ও [টেক্সট বাক্স থেকে] ইংরেজি বছরের সংখ্যা যথাক্রমে লেবেল ১৯, ১০ ও ১১ -তে কপি করে রাখা।

এর পরের ফাংশনটির কথাও ইতোমধ্যে বলেছি। এটা হলো `IsItLeap` নামক বুলিয়ান ফাংশন। এর কাজ হচ্ছে বছরটি লিপ ইয়ার কি না পরীক্ষা করে সত্য-মিথ্যা ফলাফল রিটার্ন করা। লক্ষ করুন, কোনো সংখ্যার ভাগফলে অবশিষ্ট আছে কি না জানতে হলে সিস্টেমের মড [Mod] ফাংশন ব্যবহার করতে হয়। সুতরাং এবার টা্ইপ করুন এ দু'টো প্রসিডোর।

```
Private Sub UpdateLabel() 'লেবেল আপডেট কোড।
      Label9.Text = cmbEnglishDay.SelectedItem 'কম্বো থেকে লেবেলে
       Label10.Text = cmbEnglishMonth.SelectedItem 'কম্বো থেকে লেবেলে
       Label11.Text = txtEnglishYear.Text 'টেক্সট বাক্স থেকে লেবেলে
End Sub
 'check if it is a Leap year
 Private Function IsItLeap(TheYear As Integer) As Boolean
      If TheYear Mod 4 = 0 Then 'মড রিটার্ন করে ভাগের অবশিষ্ট। না থাকলে ০
          Return True 'এটা লিপ ইয়ার
       Else
          Return False 'এটা লিপ ইয়ার নয়
      End If
```

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
End Function
```

উপরের সাব দু'টোর ব্যাখ্যা ইতোমধ্যে হয়ে গেছে। এবার পরের এই সাবটি লিখুন।

```
'এই সাব ইংরেজি আজকের তারিখ দিন, মাস ও বছর ফিল্ডে বসাবে।
Private Sub PutInEnglishDate()
    Dim now As DateTime = DateTime.Now
    Label9.Text = now.Day 'শুধু দিনের সংখ্যা
    Dim now1 As DateTime = DateTime.Now
    Label10.Text = now1.Month 'শুধু মাসের সংখ্যা
    Dim now2 As DateTime = DateTime.Now
    Label11.Text = now2.Year 'শুধু বছরের সংখ্যা
    cmbEnglishDay.SelectedItem = Label9.Text 'লেবেল থেকে কম্বো বাক্সে
    cmbEnglishMonth.SelectedItem = Label10.Text 'লেবেল থেকে কম্বো বাক্সে
    txtEnglishYear.Text = Label11.Text 'টেক্সট্ বাক্স থেকে লেবেলে
End Sub
```

উপরের সাবরুটিনটি আমরা ফর্মের লোড সাব থেকে কল করেছি। এটার কাজ হলো ইংরেজি 'আজকের' তারিখটিকে প্রথমে লুকানো লেবেল৯, লেবেল১০ ও লেবেল১১-তে যথাক্রমে দিন, মাস ও বছর সংখ্যা সংরক্ষণ করা ও কম্বোইংলিশডে, কম্বোইংলিশমান্থ ও টিএক্সটিইংলিশইয়ার বাক্সে প্রদর্শন করা। পরের সাবরুটিনের কাজ হলো ইংরেজি তারিখকে বাংলায় রূপান্তর করা।

```
 'এই সাবই কনভার্ট করে। কলস্ ফাংশন: ১.ফাইন্ডডেরেঞ্জ, ২. ফাইন্ডবাংলা ইয়ার।
Private Sub convertToBanglaDate()
    Dim EnglishDay As Integer = Val(Label9.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
   Dim EnglishMonth As Integer = Val(Label10.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
   Dim EnglishYear As Integer = Val(Label11.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
   Dim DayAdderMonth As String = FindDayRange(EnglishMonth, EnglishDay) 'ফাংশন কল
Dim Words As String() = DayAdderMonth.Split(New Char() {" "})
    ToBanglaAdderVal = Words(0)
    ToBanglaMonth = Words(1)
    ToBanglaYear = FindBanglaYear(EnglishYear, EnglishMonth, EnglishDay) 'ফাংশন কল
    Label12.Text = (ToBanglaAdderVal) + EnglishDay 'এড সঠিক মান বাংলা বছর পেতে।
    Label13.Text = ToBanglaMonth 'গোপন লেবেলে মাস মান সংরক্ষণ
    cmbBanglaDay.SelectedItem = Label12.Text 'গোপন লেবেলে থেকে কম্বোতে নেওয়া
    cmbBanglaMonth.SelectedItem = Label13.Text 'গোপন লেবেলে কম্বোতে নেওয়া
    txtBanglaYear.Text = ToBanglaYear 'বাংলা বছরের ফিল্ডে মান নেওয়া
    Label16.Text = (cmbEnglishMonth.SelectedItem & "/" & cmbEnglishDay.SelectedItem & "/" &
txtEnglishYear.Text)
    Label16.Text = FormatDateTime(Label16.Text, DateFormat.LongDate) 'দিনের নামসহ তারিখ
    Label15.Text = getBanglaDayMothName(Label16.Text, Val(Label12.Text), Val(Label13.Text), Val
(txtBanglaYear.Text)) 'দিনের নামসহ তারিখ।
End Sub
```

লক্ষ করুন, এই সাব থেকে আমরা আরো দু'টি ফাংশন কল করেছি। যথা: ১. FindDayRange এবং ২. FindBanglaYear । আশাকরি এগুলোর নাম থেকেই বুঝতে পারছেন তাদের কাজ কি। প্রথমটি রিটার্ন করে দিন একটি স্ট্রিং যার মধ্যে আছে পরিবর্তনে কোন্ সংখ্যাটি হবে এবং বাংলা মাস সংখ্যাটি। দ্বিতীয়টি রিটার্ন করে বাংলা বছর কি হবে। এই ফাংশন দু'টোর ব্যাখ্যা সঠিক স্থানে আসবে। লক্ষ করুন, আমরা Split নামক একটি সিস্টেম ফাংশন দ্বারা দু'টি সংখ্যাকে আলাদা করেছি। প্রাপ্ত দিনের মান ও মাসের মান যথাক্রমে গোপন লেবেল১২ এবং লেবেল১৩-তে নিয়ে তারপর কম্বো বাক্স দু'টোতে প্রদর্শন করেছি। অপরদিকে বছরের মানটি সরাসরি টেক্সট্‌বাক্সে নিয়ে এসেছি।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

আমাদের ইচ্ছে ব্যবহারকারী সংখ্যা হিসাবেই শুধু তারিখগুলো দেখবেন না- বরং বার ও মাসের নামসহ দেখবেন। পরের ফাংশনের কাজ হলো এটা নিশ্চিত করা। ফাংশন মোট তিনটি প্যারামিটার নেয়।

```
Public Function getBanglaDayMothName(SuplyLongDateString As String, BanglaDay As Integer, BanglaMonthVal As Integer,
BanglaYearVal As Integer) As String
    Dim AllEngDayName() As String = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"}
    Dim AllBanglaDayName() As String = {"iweevi", "†mvgevi", "g½jevi", "eyaevi", "e„n¯úwZevi", "ïµevi", "kwbevi"}
    Dim AllBanglaMonthName() As String = {"ˆekvL", "ˆRô¨", "Avlvp", "kÖveY", "fv`ª", "Avwk¦b", "KvwZ©K",
"AMÖnvqY", "†cŠl", "gvN", "dvêyb", "ˆPÎ"}
    Dim AllBangMonthsNum() As Integer = {"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12"}
    Dim RetString As String = ""
    Dim CountChar As Integer : Dim RetDay As String = "" : Dim BanglaDayname As String = ""
    Dim BanglaMonthName As String = ""
    For CountChar = 1 To Len(SuplyLongDateString)
      If Mid(SuplyLongDateString, CountChar, 1) = "," Then
        Exit For
      End If
      RetDay += Mid(SuplyLongDateString, CountChar, 1)
    Next
    For CountChar = 0 To 6
      If AllEngDayName(CountChar) = RetDay Then
        BanglaDayname = AllBanglaDayName(CountChar)
        Exit For
      End If
    Next
    For CountChar = 0 To 12
      If AllBangMonthsNum(CountChar) = BanglaMonthVal Then
        BanglaMonthName = AllBanglaMonthName(CountChar)
        Exit For
      End If
    Next
    RetString = BanglaDayname & ", " & BanglaDay & " " & BanglaMonthName & ", " & BanglaYearVal & " evsjv"
    Return RetString
  End Function
```

উপরের ফাংশন যে তিনটি প্যারামিটর নেয় এগুলো সম্পর্কে আগেই জানা দরকার। প্রথমটি হলো ইংরেজি দীর্ঘ তারিখ স্ট্রিং। এটা যে কোনো ইংরেজি তারিখ লিখে সিস্টেম দ্বারা আনা যায়। দ্বিতীয় ও তৃতীয় প্যারামিটার যথাক্রমে বাংলা দিনের সংখ্যা ও মাসের সংখ্যা। এই তিনটি ভ্যালু থেকেই বাংলা দিন ও মাসের নাম ফাংশন রিটার্ন করবে। কিভাবে? এক্ষুণি জেনে নেবেন।

প্রথমত আমরা ৪টি এ্যারে ভেরিয়েবল ডিক্লায়ার করেছি। এগুলো হয়ে: ১. AllEngDayName() যাতে আছে ইংরেজি সাত বারের নাম, ২. AllBanglaDayName() -এতে সংরক্ষিত হয়েছে বাংলা সাত বারের নাম [বিজয় কিবোর্ডে লিখেছি] ৩. AllBanglaMonthName() -এতে সংরক্ষিত আছে বাংলা বারো মাসের নাম [বিজয়ে] এবং ৪. AllBangMonthsNum() -এ ভেরিয়েবলে আছে বারো মাসের নাম্বার (১-১২)। মনে আছে এ্যারোর যাবতীয় ইলেমেন্ট সংরক্ষিত হয় ০ ইনডেক্স দ্বারা। অন্য কথায়, আমাদের প্রথম ভেরিয়েবলে আছে ০ থেকে ৬ পর্যন্ত, ইনডেক্স, দ্বিতীয়টিতেও তা-ই। তৃতীয় ও চতুর্থটিতে আছে ০ থেকে ১১ পর্যন্ত ইনডেক্স। আর আমরা এই ইনডেক্স দ্বারাই সঠিক ফলাফল বের করবো।

উপরে ব্যাখ্যাত ৪টি এ্যারে ভেরিয়েবল ছাড়াও আমরা আরো ৫টি সাধারণ ভেরিয়েবল ডিক্লায়ার করেছি ফাংশনের ভেতর কাজে লাগাতে। এদের মধ্যে ৪টির টাইপ হলো স্ট্রিং ও একটি ইন্টেজার। এগুলোর কাজ নিম্নরূপ:

১. RetString -এর কাজ বাংলা দিনের নাম, দিন সংখ্যা, মাসের নাম ও বছরের সংখ্যা একত্রে সঠিক পরম্পরায় সংরক্ষণ

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

করা। আর এটাই ফাংশন রিটার্ন করে। বাকি তিনটি স্ট্রিং ভেরিয়েবল এই প্রথমটির সহায়ক মাত্র। এগুলো দিনের নাম, মাসের নাম ও দিনের সংখ্যা সংরক্ষণ করে। ইন্টেজার ভেরিয়েবলের কাজ হলো ফর-নেক্সট লুপটি নিয়ন্ত্রণ করা।
আমরা প্রথমে একটি ফর-নেক্সট লুপ দ্বারা ইংরেজি মাসের নামটি বের করে সংরক্ষণ করেছি। এরপর এটির ফলাফলের ওপর ভিত্তি করে পরের ফর-নেস্ট লুপের মাধ্যমে সনাক্ত করেছি বাংলা দিনের নামটি। শেষের ফর-নেস্ট লুপ দ্বারা বের করেছি বাংলা মাসের নামটি। ব্যস! এবার সবগুলো একত্র করে রিটার্ন করতে হবে। এ কাজের জন্য ব্যবহার করেছি `RetString` ভেরিয়েবলকে। লক্ষ করুন কিভাবে ফরমেটিং করা হয়েছে। আমরা চাই প্রথমে হবে বাংলা দিনের নাম এরপর কমা ও স্পেইস। তারপর বাংলা দিনের সংখ্যা এরপর আরেকটি স্পেইস পরে আসবে বাংলা মাসের নাম- তারপর কমা ও স্পেইস শেষে বছরের সংখ্যা ও স্পেইস এবং সবশেষে ‘বাংলা’ শব্দটি। এই ফরমেটিং কিভাবে হয়েছে ভালো করে দেখে নেবেন। এরপর এসছে দু’টি বড়ো ফাংশনের প্রথমটি।

```
'function returns bangla day and Month to be added or subtracted
 Private Function FindDayRange(MonthInt As Integer, DayInt As Integer) As String
    If MonthInt = 1 And DayInt >= 1 And DayInt <= 13 Then
       Return "17" & " 9"
    End If
    If MonthInt = 1 And DayInt >= 14 And DayInt <= 31 Then
       Return "-13" & " 10"
    End If
    If MonthInt = 2 And DayInt >= 1 And DayInt <= 12 Then
       Return "18" & " 10"
    End If
    If MonthInt = 2 And DayInt >= 13 And DayInt <= 29 Then
       Return "-12" & " 11"
    End If
    If MonthInt = 3 And DayInt >= 1 And DayInt <= 14 Then
       Return "16" & " 11"
    End If
    If MonthInt = 3 And DayInt >= 15 And DayInt <= 31 Then
       Return "-14" & " 12"
    End If
    If MonthInt = 4 And DayInt >= 1 And DayInt <= 13 Then
       Return "17" & " 12"
    End If
    If MonthInt = 4 And DayInt >= 14 And DayInt <= 30 Then
       Return "-13" & " 1"
    End If
    If MonthInt = 5 And DayInt >= 1 And DayInt <= 14 Then
       Return "17" & " 1"
    End If
    If MonthInt = 5 And DayInt >= 15 And DayInt <= 31 Then
       Return "-14" & " 2"
    End If
    If MonthInt = 6 And DayInt >= 1 And DayInt <= 14 Then
       Return "17" & " 2"
    End If
    If MonthInt = 6 And DayInt >= 15 And DayInt <= 30 Then
       Return "-14" & " 3"
    End If
    If MonthInt = 7 And DayInt >= 1 And DayInt <= 15 Then
       Return "16" & " 3"
    End If
    If MonthInt = 7 And DayInt >= 16 And DayInt <= 31 Then
```

ব্যাখ্যা: এই ফাংশনটি বেশ বড়ো সত্যি, কিন্তু এটি একটি মাত্র স্ট্রিং ভ্যালু রিটার্ন করে, যার মধ্যে আছে আমাদের কাঙ্ক্ষিত দু’টি সংখ্যা। ফাংশন দু’টি প্যারামিটার গ্রহণ করে। এগুলো হলো: ইংরেজি মাসের সংখ্যা [১-১২] এবং দিনের সংখ্যা [১-৩১]। এ দু’টো সংখ্যা কি কি তার ওপর নির্ভর করে বাংলা দিনের সংখ্যা বের করতে কোন্ সংখ্যাটি যোগ বা বিয়োগ দিতে হবে এবং বাংলা মাসের সংখ্যাটি কি হবে। যেমন মাস জানুয়ারী [১] হলে এবং দিন ১ থেকে ১৩ হলে বাংলা দিন সংখ্যা বানাতে ১৭ যোগ দিতে হবে। অনুরূপ মাস সংখ্যা যদি ৪ হয় আর দিনের সংখ্যা ১৪ থেকে ৩০ থাকে, তাহলে বাংলা দিন বানাতে বিয়োগ দিতে হবে ১৩ ইত্যাদি। ফাংশন একে একে সবগুলো কম্বিনেশন চেক করে সঠিক যোগ বা বিয়োগের সংখ্যাটি ও মাসের মানটি কলিং সাবে রিটার্ন করে। পূর্বে আমরা যে সম্পর্কের টেবিলটি দেখেছিলাম সে’টি কাজে লাগিয়েই এই ফাংশনটি তৈরি করেছি। লক্ষ করুন, বিয়োগ বুঝাতে সংখ্যার পূর্বে একটি বিয়োগ চিহ্ন দেওয়াই যথেষ্ট।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
        Return "-15" & " 4"
    End If
    If MonthInt = 8 And DayInt >= 1 And DayInt <= 15 Then
        Return "16" & " 4"
    End If
    If MonthInt = 8 And DayInt >= 16 And DayInt <= 31 Then
        Return "-15" & " 5"
    End If
    If MonthInt = 9 And DayInt >= 1 And DayInt <= 15 Then
        Return "16" & " 5"
    End If
    If MonthInt = 9 And DayInt >= 16 And DayInt <= 30 Then
        Return "-15" & " 6"
    End If
    If MonthInt = 10 And DayInt >= 1 And DayInt <= 15 Then
        Return "15" & " 6"
    End If
    If MonthInt = 10 And DayInt >= 16 And DayInt <= 31 Then
        Return "-15" & " 7"
    End If
    If MonthInt = 11 And DayInt >= 1 And DayInt <= 14 Then
        Return "16" & " 7"
    End If
    If MonthInt = 11 And DayInt >= 15 And DayInt <= 30 Then
        Return "-14" & " 8"
    End If
    If MonthInt = 12 And DayInt >= 1 And DayInt <= 14 Then
        Return "16" & " 8"
    End If
    If MonthInt = 12 And DayInt >= 15 And DayInt <= 31 Then
        Return "-14" & " 9"
    End If
    Return 0
End Function
```

ইতোমধ্যে যে ফাংশনটির ওপর বিস্তারিত বলা হয়েছে তা থেকে আমরা বাংলা দিন ও মাস পাবো মাত্র। তবে আমাদেরকে তো বাংলা বছর কি হবে তা-ও জানতে হবে। এটা জানার জন্য পরের এই ফাংশনটি।

```
Private Function FindBanglaYear(EngYear As Integer,
EngMonth As Integer, DateDay As Integer) As Integer
    If EngMonth >= 1 And EngMonth <= 3 Then
        ToBanglaYear = EngYear - 594 'because
banglaYear 1 less, no Boishak yet
        Return ToBanglaYear
    End If
    If EngMonth = 4 And DateDay >= 1 And
DateDay <= 13 Then
        ToBanglaYear = EngYear - 594 'because
banglaYear 1 less, no Boishak yet
        Return ToBanglaYear
    End If
    ToBanglaYear = EngYear - 593 'Here it change
    Return ToBanglaYear
End Function
```

লক্ষ করুন, উপরের ফাংশনে তিনটি প্যারামিটার আছে। এগুলো হলো যথাক্রমে ইংরেজি বছর, মাস ও দিন। প্রতিটি ভ্যালু হবে ইন্টেজার। ফাংশন রিটার্ন করবে বাংলা বছর [ইন্টেজার হিসাবে]। আমরা জানি ১লা বৈশাখ না আসা পর্যন্ত বাংলা বছর সংখ্যা বাড়বে না। এ কারণেই প্রথমে ইংরেজি মাস ১ থেকে ৩ এর ভেতর হলে বাংলা মাস বিয়োগ করেছি ৫৯৪ সংখ্যা দ্বারা। বাকি মাসগুলোর ক্ষেত্রে বিয়োগ হবে ৫৯৩।

আমাদের ফর্মে উপরের বায়ের বোতামটি হলো বর্তমান তারিখ বাংলা ও ইংরেজি তারিখ প্রর্দনের বোতাম। সুতরাং এটার ক্লিক ইভেন্ট হবে নিম্নরূপ।

```
'English to Bangla Date conversion button sub
Private Sub Button2_Click(sender As Object, e As
EventArgs) Handles Button2.Click
    Label15.Text = ""
    PutInEnglishDate() 'পরিবর্তন [ইংরেজি থেকে বাংলা]
বোতাম। কল সাব।
    convertToBanglaDate() 'কল সাব পরিবর্তন করতে।
End Sub
```

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

এই সাবে তিনটি মাত্র লাইন আছে। প্রথমে মুছে দিয়েছি লেবেল১৫। এ লেবেলে নতুন লেখা আসবে। এর পরের দু'টি লাইন হলো দু'টি সাবরুটিন। এ উভয়টির ব্যাখ্যা ইতোমধ্যে হয়েছে। পরের ফাংশনের কাজ হলো ইংরেজি দিন সংখ্যাটি মাসের সাথে সামঞ্জস্যশীল আছে কি না তা পরীক্ষা করা। মনে করুন ব্যবহারকারী মাস ৯ [সেপ্টেম্বর] প্রবেশ করলেন। এখন দিনের ফিল্ডে যদি ৩১ প্রবেশ করেন তাহলে তো সমস্যা দাঁড়াবে। কারণ, সেপ্টেম্বর মাস ৩০ দিনের বেশি হয় না। এই সমস্যা নিরসনে নিচের ফাংশনটি কাজ করে। মাসের মান অনুযায়ী দিনের মান না থাকলে ফাংশন মিথ্যা ফেরৎ পাঠাবে। অন্যথায় সত্য।

```
 'Day entry correct: Returns True, Else False
Private Function IsDayEntryCorrect(Month As Integer, Day As Integer) As Boolean
      If Month = 9 And Day > 30 Then
         Return False
      End If
      If Month = 4 And Day > 30 Then
         Return False
      End If
      If Month = 6 And Day > 30 Then
         Return False
      End If
      If Month = 11 And Day > 30 Then
         Return False
      End If
      Return True
End Function
```

এখন আমরা বাংলা থেকে ইংরেজি তারিখ পরিবর্তন বোতামের ক্লিক ইভেন্ট কোড দেখবো।

```
'Bangla to English Date conversion button sub
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
      If Val(txtBanglaYear.Text) < -493 Or Val(txtBanglaYear.Text) > 9999 Then
         MsgBox("বছর রেঞ্জ হবে -৪৯৩ থেকে ৯৯৯৯।", MsgBoxStyle.Exclamation) 'রেঞ্জের বাইরে।
         txtBanglaYear.Text = ""
         txtBanglaYear.Focus()
         Exit Sub
      End If
      Label15.Text = ""
      UpdateBLabel() 'take day, month and year values to hidden labels
      If IsBanglaDateCorrect(Val(Label12.Text), Val(Label13.Text), Val(Label14.Text)) = False Then
         MsgBox("উপস্ তারিখ সঠিক হয় নি! যে মাসে যতোদিন আছে তারচেয়ে বেশি হয়েছে, সম্ভবত।",
MsgBoxStyle.Exclamation)
      End If
      convertToEnglishDate()
End Sub
```

প্রথমে বছরের রেঞ্জ সঠিক আছে কি না পরীক্ষা করে [সঠিক থাকলে] আমরা UpdateBLabel() সাব কল করে দিন, মাস ও বছরের মানগুলো গোপন লেবেলে নিয়ে সংরক্ষণ করেছি। এরপর আবারো তরিখ সঠিক কি না পরীক্ষা করে [সঠিক হলে] ডাক দিয়েছি পরিবর্তন করার মূল সাবরুটিন convertToEnglishDate()। পরের সাবটি হলো UpdateBLabel() -এর সাবরুটিন। এটার কাজ কি তা ইতোমধ্যে বলেছি।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
'update Bagla label
Private Sub UpdateBLabel() 'লেবেল আপডেট কোড।
      Label12.Text = cmbBanglaDay.SelectedItem 'কম্বো থেকে লেবেলে
      Label13.Text = cmbBanglaMonth.SelectedItem 'কম্বো থেকে লেবেলে
      Label14.Text = txtBanglaYear.Text 'টেক্সট বাক্স থেকে লেবেলে
End Sub
```

আমরা ইতোমধ্যে বলেছি, বাংলা তারিখ ঠিক আছে কি না পরীক্ষার জন্য একটি ফাংশন আছে। নিচের কোডগুলোই হচ্ছে এ ফাংশন। এটি লিপ ইয়ার ও সঠিক তারিখ, মাসের সঙ্গে তারিখের সামঞ্জস্য ইত্যাদি পারীক্ষা করে সত্য বা মিথ্যা ফলাফল হিসাবে রিটার্ন করে।

```
'check to see if the dateEntry correct
Private Function IsBanglaDateCorrect(BDay As Integer, BMonth As Integer, BYear As Integer) As Boolean
    Dim CheckLeap As Boolean = IsItLeap(BYear)
    If CheckLeap = True Then
      If BMonth = 11 And BDay <= 31 Then
        Return True
      End If
      If BMonth >= 1 And BMonth <= 5 Then
        If BDay <= 31 Then
          Return True
        End If
      End If
      If BMonth >= 6 And BMonth <= 10 Then
        If BDay <= 30 Then
          Return True
        End If
      End If
      If BMonth = 12 And BDay <= 30 Then
        Return True
      End If
    End If
    'leap year false
    If BMonth >= 1 And BMonth <= 5 Then
      If BDay <= 31 Then
        Return True
      End If
    End If
    If BMonth >= 6 And BMonth <= 11 Then
      If BDay <= 30 Then
        Return True
      End If
    End If
    If BMonth = 12 And BDay <= 30 Then
      Return True
    End If
    Return False
End Function
```

পরের সাবরুটিনের কাজ হলো বাংলা তারিখকে ইংরেজি তারিখে রূপান্তর করা।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
'bangla to english date conversion
Private Sub convertToEnglishlDate()
    Dim BanglaDay As Integer = Val(Label12.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim BanglaMonth As Integer = Val(Label13.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim BanglaYear As Integer = Val(Label14.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim DayAdderMonth As String = FindEnglishRange(BanglaMonth, BanglaDay) 'ফাংশন কল
    Dim Words As String() = DayAdderMonth.Split(New Char() {" "})
    ToEnglishAdderVal = Words(0)
    ToEnglishMonth = Words(1)
    ToEnglishYear = FindEnglishYear(BanglaYear, BanglaMonth, BanglaDay) 'ফাংশন কল
    Label9.Text = BanglaDay + (ToEnglishAdderVal)  'এড সঠিক মান বাংলা দিন পেতে।
    Label10.Text = ToEnglishMonth 'গোপন লেবেলে মাস মান সংরক্ষণ
    cmbEnglishDay.SelectedItem = Label9.Text 'গোপন লেবেল থেকে কম্বোতে নেওয়া
    cmbEnglishMonth.SelectedItem = Label10.Text 'গোপন লেবেল থেকে কম্বোতে নেওয়া
    txtEnglishYear.Text = ToEnglishYear 'বাংলা বছরের ফিল্ডে মান নেওয়া
    Label16.Text = cmbEnglishMonth.SelectedItem & "/" & Label9.Text & "/" & txtEnglishYear.Text
    Label16.Text = FormatDateTime(Label16.Text, DateFormat.LongDate)
    Label15.Text = getBanglaDayMothName(Label16.Text, Val(Label12.Text), Val(Label13.Text), Val
    (Label14.Text)) 'দিনের নামসহ তারিখ
End Sub
```

এই সাবরুটিন দু'টি গুরুত্বপূর্ণ ফাংশন কল করে। এগুলো সম্পর্কে ব্যাখ্যা একটু পরই আসছে। এখানে যা বলার তাহলো, প্রথম ফাংশন রিটার্ন করে একটি স্ট্রিংয়ের মধ্যে দু'টি মান। প্রথমটি হলো ইংলিশ দিনের সাথে কতো যোগ বা বিয়োগ করতে হবে। আর দ্বিতীয়টি হলো বাংলা মাসটি কি হবে। আমরা স্ট্রিংকে সিস্টেমের Split নামক ফাংশন দ্বারা আলাদা দু'টি শব্দে বিভক্ত করে মানগুলো বের করে যথাক্রমে ToEnglishAdderVal এবং ToEnglishMonth ভেরিয়েবলে সংরক্ষণ করেছি। দ্বিতীয় ফাংশন থেকে প্রাপ্ত ইংলিশ বছরের মান সরাসরি টেক্সবাক্সে নিয়ে গেছি প্রদর্শনের জন্য। লক্ষ করুন টেক্সট বাক্স ১৫ ও ১৬-তে দিনের ও মাসের নামসহ যথাক্রমে ইংরেজি ও বাংলা তারিখটি বসিয়েছি। ঠিক যেমনটি করেছিলাম ইংরেজি থেকে বাংলায় তারিখ পরিবর্তনের সময়। এবার আরো একটি ফাংশনের কোড দেখবো।

```
'returns bangla year
Private Function FindEnglishYear(BangYear As Integer, BangMonth As Integer, DateDay As Integer) As Integer
    If BangMonth >= 1 And BangMonth <= 8 Then
        ToEnglishYear = BangYear + 593 'because banglaYear 1 less, no Boishak yet
        Return ToEnglishYear
    End If
    If BangMonth = 9 And DateDay >= 1 And DateDay <= 17 Then
        ToEnglishYear = BangYear + 593 'because banglaYear 1 less, no Boishak yet
        Return ToEnglishYear
    End If
    ToEnglishYear = BangYear + 594 'Here it change
    Return ToEnglishYear
End Function
```

আমাদেরকে ইংরেজি বছরের মান বের করতে হবে। উপরের ফাংশনের কাজ এটি। লক্ষ করুন, বাংলা মাস যদি ১ থেকে ৪ এর ভেতর হয় তাহলে যোগ করতে হবে ৫৯৩। একইভাবে বাংলা মাস ৯ [ফাল্গুন] হলে ও দিনের মান ১ থেকে ১৭ পর্যন্ত হলে যোগ হবে ৫৯৩। অন্যথায় যোগ হবে ৫৯৪।

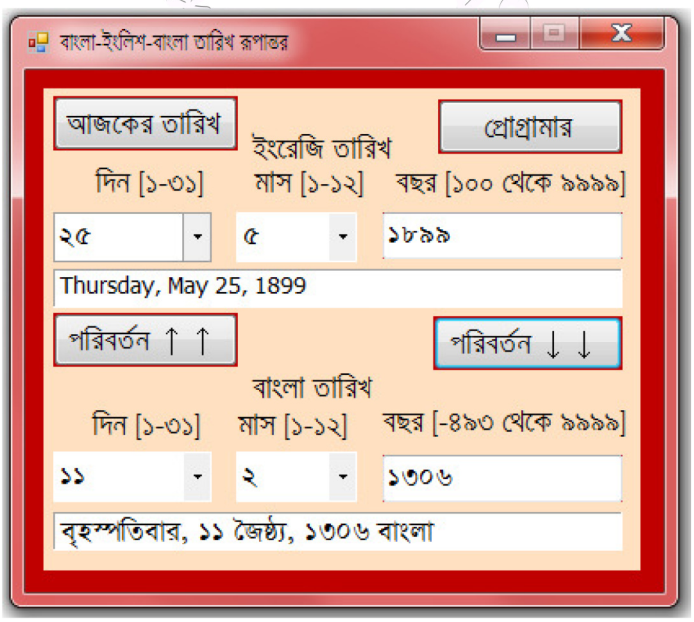
১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

এবার যে ফাংশনটি দেখবো তা-ই প্রোগ্রামের সর্বশেষ কোড ব্লক। অবশ্য ডানের উপরের বোতামের কোডও আরেকটি আছে। এটা প্রোগ্রামার সম্পর্কে জ্ঞাতব্য ব্যাপার মাত্র।

```
'returns what number to be added [or subtracted] to get correct english day value
Private Function FindEnglishRange(MonthInt As Integer, DayInt As Integer) As String
    If MonthInt = 1 And DayInt >= 1 And DayInt <= 17 Then
      Return "13" & " 4"
    End If
    If MonthInt = 1 And DayInt >= 18 And DayInt <= 31 Then
      Return "-17" & " 5"
    End If
    If MonthInt = 2 And DayInt >= 1 And DayInt <= 17 Then
      Return "14" & " 5"
    End If
    If MonthInt = 2 And DayInt >= 18 And DayInt <= 31 Then
      Return "-17" & " 6"
    End If
    If MonthInt = 3 And DayInt >= 1 And DayInt <= 16 Then
      Return "14" & " 6"
    End If
    If MonthInt = 3 And DayInt >= 17 And DayInt <= 31 Then
      Return "-16" & " 7"
    End If
    If MonthInt = 4 And DayInt >= 1 And DayInt <= 16 Then
      Return "15" & " 7"
    End If
    If MonthInt = 4 And DayInt >= 17 And DayInt <= 31 Then
      Return "-16" & " 8"
    End If
    If MonthInt = 5 And DayInt >= 16 And DayInt <= 15 Then
      Return "15" & " 8"
    End If
    If MonthInt = 5 And DayInt >= 17 And DayInt <= 31 Then
      Return "-16" & " 9"
    End If
    If MonthInt = 6 And DayInt >= 1 And DayInt <= 15 Then
      Return "15" & " 9"
    End If
    If MonthInt = 6 And DayInt >= 16 And DayInt <= 30 Then
      Return "-15" & " 10"
    End If
    If MonthInt = 7 And DayInt >= 1 And DayInt <= 16 Then
      Return "15" & " 10"
    End If
    If MonthInt = 7 And DayInt >= 17 And DayInt <= 30 Then
      Return "-16" & " 11"
    End If
    If MonthInt = 8 And DayInt >= 1 And DayInt <= 16 Then
      Return "14" & " 11"
    End If
    If MonthInt = 8 And DayInt >= 17 And DayInt <= 30 Then
      Return "-16" & " 12"
    End If
    If MonthInt = 9 And DayInt >= 1 And DayInt <= 17 Then
      Return "14" & " 12"
    End If
    If MonthInt = 9 And DayInt >= 18 And DayInt <= 30 Then
      Return "-17" & " 1"
    End If
    If MonthInt = 10 And DayInt >= 1 And DayInt <= 18 Then
      Return "13" & " 1"
    End If
    If MonthInt = 10 And DayInt >= 19 And DayInt <= 30 Then
      Return "-18" & " 2"
    End If
    If MonthInt = 11 And DayInt >= 1 And DayInt <= 17 Then
      Return "12" & " 2"
    End If
    If MonthInt = 11 And DayInt >= 17 And DayInt <= 31 Then
      Return "-16" & " 3"
    End If
    If MonthInt = 12 And DayInt >= 1 And DayInt <= 17 Then
      Return "14" & " 3"
    End If
    If MonthInt = 12 And DayInt >= 18 And DayInt <= 30 Then
      Return "-17" & " 4"
    End If
    Return 0
End Function
```

এখানেই ফাংশনের শেষ। এ'টি কাজ করে আগে বর্ণিত FindDayRange নামক ফাংশনের মতো। ফাংশনের দু'টি প্যারামিটার আছে। তাহলো, ইংরেজি মাসের মান ও দিনের মান। কোন্ মাস ও কোন দিন তা-ই নির্দিষ্ট করে বাংলা কোন্ মাস ও দিন হবে। আগেরটির মতো এই ফাংশনও একটি স্ট্রিং রিটার্ন করে যেখানে দু'টি সংখ্যা থাকবে। আমরা কলিং সাবরুটিনে Split নামক সিস্টেম ফাংশন দ্বারা আলাদা করেছি। এরপর উভয় সংখ্যা কাজে লাগিয়েছি। যাক, কোডের ব্যাখ্য এখানেই শেষ। এবার রান করে দেখুন [নিচের চিত্র]।



১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

আপনি নিশ্চয়ই এ গ্রন্থ অধ্যয়নকালে Visual Studio Express For Desktop [2008, 2010, 2012, 2013 or 2015] এর যে কোনো একটি ব্যবহার করছেন । ভার্শন যেটাই ব্যবহার করুন না কেনো, গ্রন্থে উল্লেখিত সকল কোডই যে কোনোটির জন্য প্রজোয্য। তবে আমি কোডগগুলো লিখেছি ভার্শন ২০১২ দ্বারা। প্রোগ্রামারদের সুবিধার জন্য মাইক্রসফট উক্ত প্রতিটি ভার্শনের মধ্যেই [সামান্য রদবদলসহ] একটি উন্নতমানের টুল্‌স বাক্স সম্পৃক্ত করেছেন। আপনারা দেখেছেন এ টুলস্‌বাক্স থেকে কিভাবে আমার বোতাম, টেক্সটবাক্স, কম্বোবাক্স ইত্যাটি বস্ত ডবলক্লিক বা টেনে টেনে ফর্মে নিয়ে এসেছি। এরপর এদের প্রোপার্টি নিজের চাহিদামাফিক পরিবর্তন করেছি। এই টুল্‌স বাক্সের মধ্যে যা কিছু আছে তার সবগুলো আমার ব্যবহার কখনো করি নি। তবে এগুলোর মৌলিক কার্যকারিতা নিয়ে এ পর্যন্ত বিস্তারিত আলোচনা হয় নি। এ পরিচ্ছেদ রচনার উদ্দেশ্য হলো টুল্‌স বাক্সের বস্ত বা অবজেক্টের উপর তথ্যাদি তুলে ধরা। প্রয়োজনে ছোট ছোট কোড্‌ উদাহরণও থাকবে। আমি উপদেশ দেবো, আপনি প্রথমে একটি উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করুন। ফর্মের আয়তন বড়ো করুন। যেসব কোড নিয়ে আলোচনা হবে ওগুলো বাস্তবে লিখে প্রোগ্রাম চালিয়ে দেখুন- এতে কোড ও বস্তুটি সম্পর্কে অতিরিক্ত জানা হয়ে যাবে।

যাক, প্রথমে আমরা টুল্‌স বাক্সের পুরো অবজেক্ট লিস্ট তুলে ধরছি। এরপর একে একে ব্যাখ্যা হবে।

1. Pointer
2. BackgroundWorker
3. BindingNavigator
4. BindingSource
5. Button
6. CheckBox
7. CheckedListBox
8. ColorDialog
9. ComboBox
10. ContextMenuStrip
11. DataGridView
12. DataSet
13. DateTimePicker
14. DirectoryEntry
15. DirectorySearcher
16. DomainUpDown
17. ErrorProvider
18. EventLog
19. FileSystemWatcher
20. FontDialog
21. GroupBox
22. HelpProvider
23. HScrollBar
24. ImageList
25. Label
26. LinkLabel
27. ListBox
28. ListView
29. MaskedTextBox
30. MenuStrip
31. MessageQueue
32. MonthCalender
33. NotifyIcon
34. NumericUpDown
35. OpenFileDialog
36. PageSetupDialog
37. Panel
38. PerformanceCounter
39. PictureBox
40. PrintDialog
41. PrintDocument
42. PrintPreviewControl
43. PrintPreviewDialog
44. Process
45. ProgressBar
46. PropertyGrid
47. RadioDutton
48. RicjTextBox
49. SaveFileDialof
50. SerialPort
51. ServiceController
52. SplitContainer
53. Splitter
54. StatusStrip
55. TabControl
56. TableLayoutPanel
57. TextBox
58. Timer
59. ToolStrip
60. ToolStripContainer
61. ToolTip
62. TrackBar
63. TreeView
64. VScrollBar
65. WebBrowser
66. Chart [Inside Data Sub Folder]

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

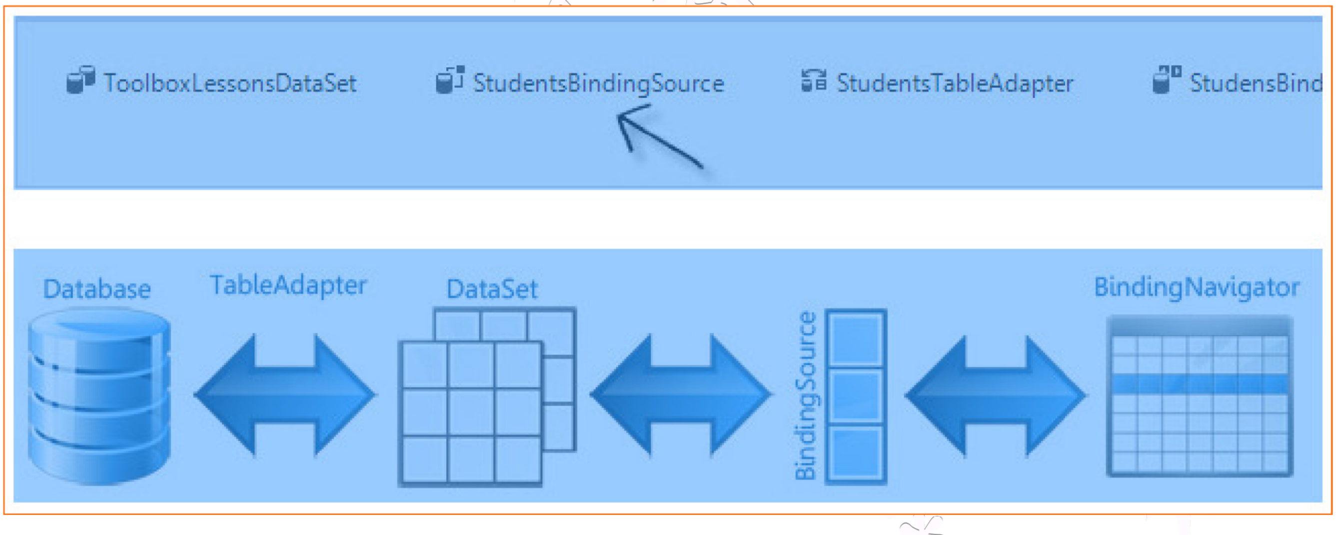
1. Pointer: টুল্স বাক্সের প্রথম সদস্য হলেও এটা বাস্তবে কোনো কন্ট্রোল নয়! তবে ফর্মে কন্ট্রোল নেওয়ার জন্য এটি কাজ করে। এছাড়া সিলেক্ট করা কোনো কন্ট্রোলকে ডিসিলেক্ট করতেও এটি ব্যবহৃত হয়।

2. BackgroundWorker: এটা বেকগ্রাউন্ড বা নেপথ্যে কাজ করে- কিন্তু ফর্মের ওপর এর কোনো এফেক্ট নেই। অথচ ফর্মের কোড থেকেই একে নির্দেশনা দেওয়া যায়। দীর্ঘ কোনো অপারেশনে বেকগ্রাউন্ডওয়ার্কার খুব ব্যবহারোপযোগী।

3. BindingNavigator: এটা ডাটা রেকর্ডের মধ্যে ন্যাভিগেট করার জন্য ব্যবহৃত হয়। আমরা যখন ভিজুয়্যাল স্টুডিও এর উইজার্ড দ্বারা ডাটাবেইজ প্রোগ্রাম সৃষ্টি করেছিলাম, তখন এমনিতেই উপরে কয়েকটি ন্যাভিগেশন বোতাম এসেছিল। এগুলো এই বাইন্ডিংন্যাভিগেটরের কাজ। এটি ফর্মের নিচে ধূসর এলাকায় পড়ে থাকে। চিত্রটি দেখুন।

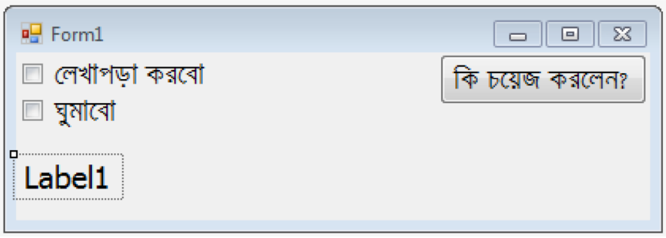


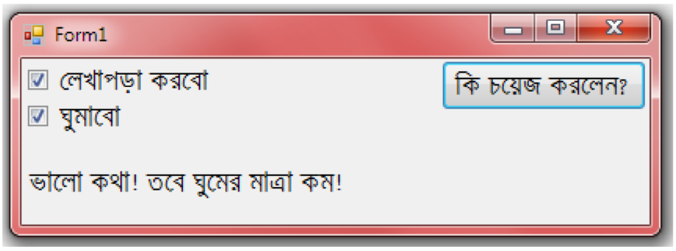
4. BindingSource: এটার কাজ হলো ফর্মের মধ্যে সম্পর্কিত ডাটা নিয়ন্ত্রণ করা। ভিজুয়্যাল বেসিক.নেট ডাটাবেইজের ডাটা প্রদর্শন ও নিয়ন্ত্রণে সাধারণত চারটি অবজেক্ট বা কন্ট্রোল ব্যবহার করে: 1. DataSet, 2. TableAdapter, 3. BindingSource & 4. BindingNavigator। দেখুন নিচের ডায়াগ্রামটি।



5. Button: এটা সম্পর্কে বেশি বলার প্রয়োজন নেই- শুধু এটুকু ছাড়া: প্রায় সকল এ্যাপলিকেশনেই এই অবজেক্টটি ব্যবহৃত হয়। এ গ্রন্থের প্রতিটি উদাহরণে এক বা ততোধিক বোতাম ব্যবহৃত হয়েছে।

6. CheckBox: এ বস্তুটি আমরা খুব বেশি ব্যবহার করি নি। এটা মূলত একটি বাক্স যেখানে ব্যবহারকারী ক্লিক করলে একটি চেকমার্ক আসে। পুনরায় ক্লিক করলে তা চলে যায়। বাক্সের ডানে [আপনার নিজস্ব] যা লেখা থাকবে তা অত্যন্ত গুরুত্বপূর্ণ। যেমন আপনি লিখতে পারেন: Volume এবং আরেকটিতে লিখতে পারেন: Area। এখন ব্যবহারকারী উভয়টি কিংবা যে কোনো একটি কিংবা কোনোটি নয়- এভাবে চেকমার্ক করতে পারেন। এ থেকে আপনি জানলেন কোন্‌টি চয়েজ করা হয়েছে ও প্রয়োজনীয় একশন নিলেন কোডের মাধ্যমে। একটি ছোট্ট উদাহরণ দেখুন। বায়ের চিত্রে

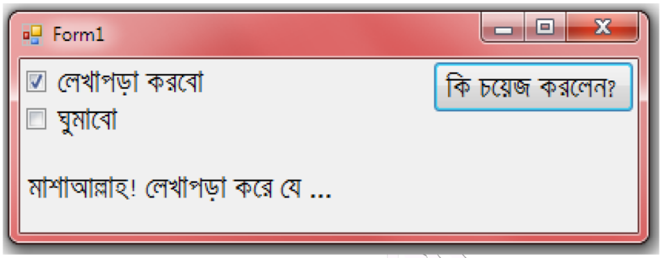


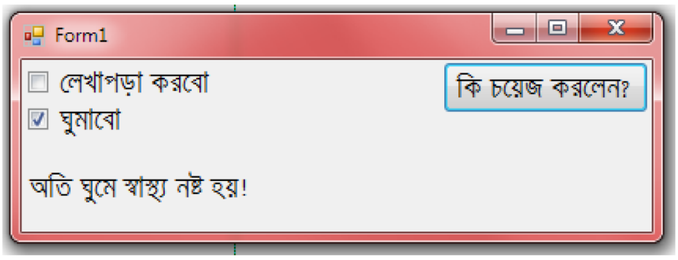


১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

ডিজাইন তুলে ধরেছি ডানেরটিতে উভয় বোতাম চয়েজ করার পর কী মেসেজ লেবেলে আসবে তা দেখা যাচ্ছে। বোতামের কোডে আরো দু'টি চয়েজের জন্য দু'টি ভিন্ন মেসেজও আছে। নিচের চিত্রদ্বয় দেখুন।





ফর্মের 'কি চয়েজ করলেন?' বোতামের ক্লিক ইভেন্ট সাবে লিখুন:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    If CheckBox1.Checked = True Then
        Label1.Text = "মাশাআল্লাহ! লেখাপড়া করে যে ..."
    End If
    If CheckBox2.Checked = True Then
        Label1.Text = "অতি ঘুমে স্বাস্থ্য নষ্ট হয়!"
    End If
    If CheckBox1.Checked = True And CheckBox2.Checked = True Then
        Label1.Text = "ভালো কথা! তবে ঘুমের মাত্রা কম!"
    End If
    If CheckBox1.Checked = False And CheckBox2.Checked = False Then
        Label1.Text = "আরে! ঘুমাবেনও না, পড়বেনও না! মনে হয় আপনি লেজি!!"
□□□□□□□□End If
End Sub
```

7. CheckedListBox: এটা উপরে CheckBox ব্যাখ্যাকৃত কন্ট্রোলের মতোই কাজ করে। তবে অবশ্যই কয়েকটি মৌলিক পার্থক্য আছে। একটি উদাহরণ দেখি।

ফর্মের মধ্যে একটি CheckedListBox control যোগ করুন। কন্ট্রোলের উপর রাইট ক্লিক করে চয়েজ করুন: Edit Items ...। একটি জানালা আসবে এতে লিস্ট লিখুন:

□&(□□) □□ □□□□

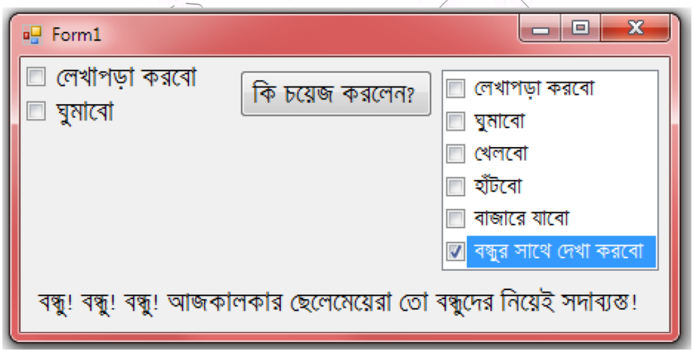
*□□□□□

□&(□□□

□□+□□□□□□□□

□, □□$ □□+ □□'(□□ □□□□

□



Ok বোতামে ক্লিক করুন। দেখবেন লিস্টটি এসে গেছে।

এবার নিচের কোড ব্লকটি চেকড্‌লিস্টবাক্সের মধ্যে ক্লিক করার পর যে সাবরুটিন আসবে তার ভেতর লিখুন। রান করুন, টেস্ট করুন।

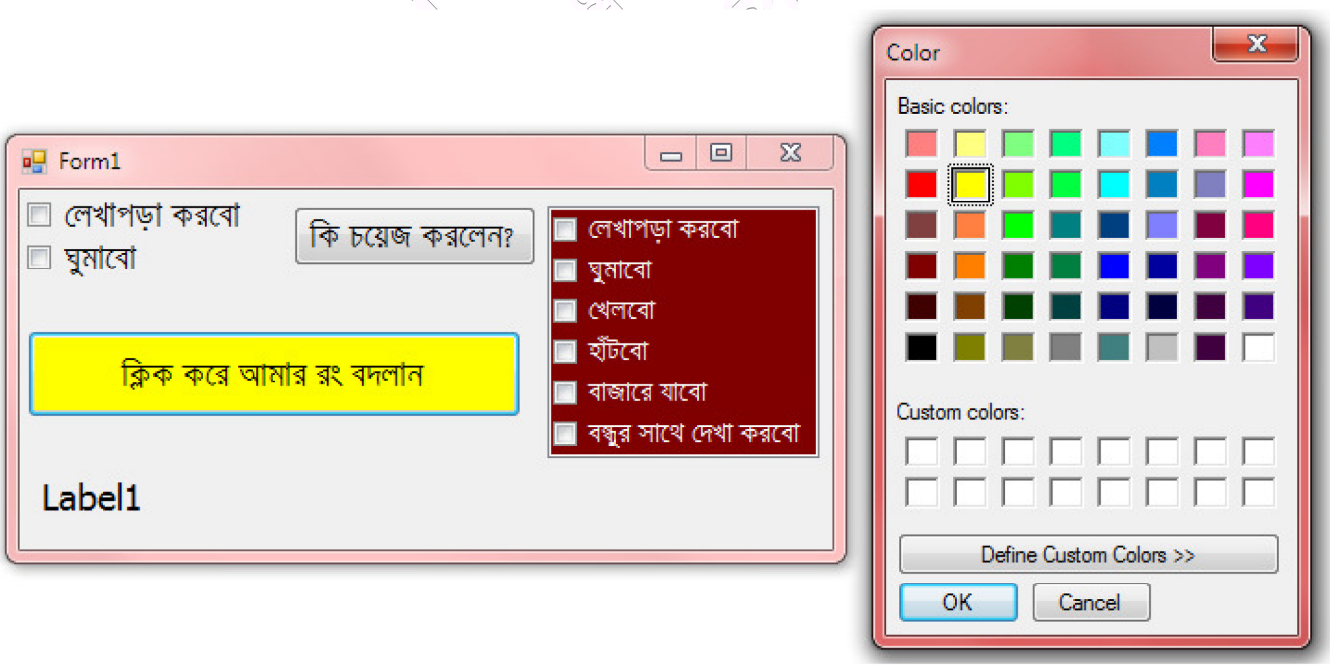
```
Private Sub CheckedListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
CheckedListBox1.SelectedIndexChanged
    With CheckedListBox1
        If .SelectedIndex = 0 Then
            Label1.Text = "... গাড়ি ঘোড়া চড়বে! জ্ঞানী হবে।"
                End If
        If .SelectedIndex = 1 Then
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
            Label1.Text = "ঠিক আছে! সাবইকেই ঘুমোতে হয়- তবে অতিরিক্ত নয়!"
        End If
        If .SelectedIndex = 2 Then
            Label1.Text = "খেলাধুলো শরীরের জন্য ভালো বটে! কিন্তু বেশি না!"
        End If
        If .SelectedIndex = 3 Then
            Label1.Text = "অকে! হাঁটলে ব্যায়াম হয়। ব্যায় হলে স্বাস্থ্য ভালো থাকে!"
        End If
        If .SelectedIndex = 4 Then
            Label1.Text = "খুব তাড়াতাড়ি ফিরে এসো! বধুয়া অপেক্ষা করিতেছে, রাঁধিবে!"
        End If
        If .SelectedIndex = 5 Then
            Label1.Text = "বন্ধু! বন্ধু! বন্ধু! আজকালকার ছেলেমেয়েরা তো বন্ধুদের নিয়েই সদাব্যস্ত!"
        End If
    End With
End Sub
```

8. ColorDialog: এটা হচ্ছে রং করার ডাইলোগ প্রদর্শনের কন্ট্রোল। নিচের উদাহরণটি দেখুন। উপরের ফর্মে একটি বোতাম ও একটি কালার ডায়লগ কন্ট্রোল টুলবাক্স থেকে যোগ করুন। বোতামের ক্লিক ইভেন্ট সাবরুটিনে লিখুন:



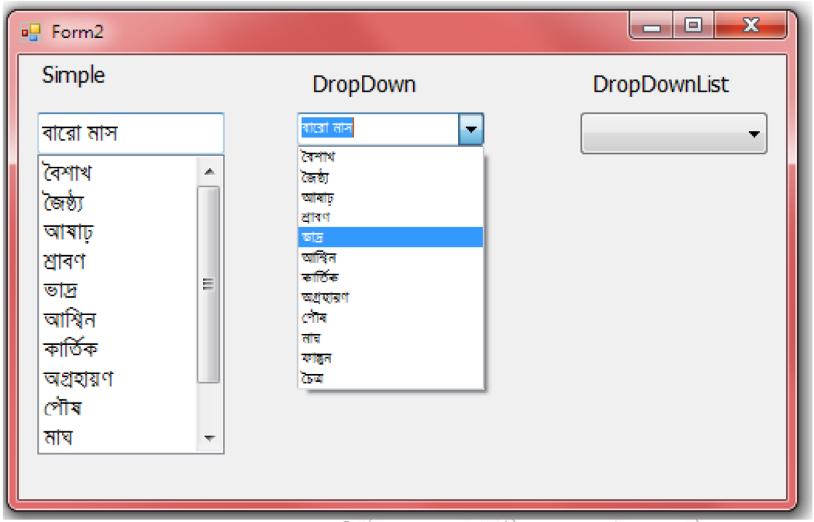

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    If ColorDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
        Button2.BackColor = ColorDialog1.Color
    End If
End Sub
```

উপরের চিত্রে রানকালীন অবস্থা তুলে ধরেছি। ডানের ডায়লগটিই হলো **ColorDialog**। আপনি যে কোনো রঙে ক্লিক করে OK দিন।

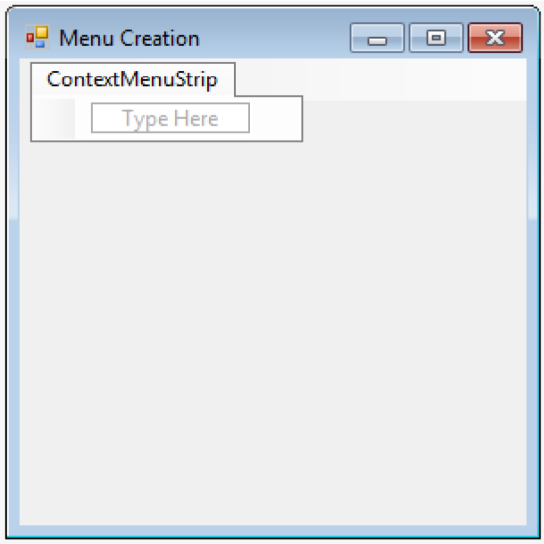
১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

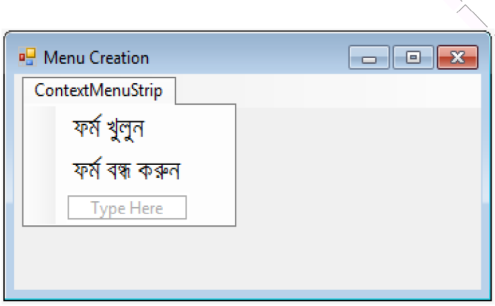
9. ComboBox: এটার ব্যবহার আমরা দেখেছি। তিন ধরনের কম্বোবক্স আছে: Simple, DropDown and DropDownList।



বাম়ের চিত্রে তিন ধরনের কম্বোবাক্স দেখা যাচ্ছে। লক্ষণীয় যে, টুলবাক্সে কম্বোবাক্স একটাই। কিন্তু প্রোপার্টিতে এর ধরন নির্ধারণ করা যায়। এই প্রোপার্টির নাম DropDownStyle। এখানেই তিনটির যে কোনো একটি চয়েজ করতে পারেন।



10. ContextMenuStrip: এটা মাউসে রাইট-ক্লিক [Right-Click] করে মেনু তৈরির অবজেক্ট। ফর্মে নিলেই এটা মেনু তৈরির [ডিজাইনের] জন্য প্রস্তুতি নেবে। পাশের চিত্রটি দেখুন। এরপর টাইপ করে আপনার মেনু আইটেম যোগ করতে পারেন। লক্ষ করুন, আমাদের উদাহরণে বাংলায় মেনু আইটেম লিখেছি। ইউনিকোডে লিখলে কোনো সমস্যা নেই। এ উভয় মেনু আইটেমের ক্লিক ইভেন্ট হ্যান্ডলারে কোড লিখে আমরা নির্দেশ দিতে পারি। আমাদের উদ্দেশ্য ব্যবহারকারী প্রথম মেনু আইটেম চয়েজ করলে একটি ফর্ম খুলবে। দ্বিতীয়টি দ্বারা তা বন্ধ হয়ে যাবে। প্রথমে আমাদেরকে অবশ্য সেই ফর্মটি প্রজেক্টে যুক্ত করতে হবে। সুতরাং প্রজেক্ট মেনু থেকে একটি নতুন ফর্ম যুক্ত করুন। এবার এই কোডগুলো প্রথম ফর্মের ক্লাসের ভেতর লিখুন। মনে রাখবেন ফর্মের সাথে এই মেনুকে জড়িত করতে হবে। এটা করতে ফর্মে ক্লিক করুন। প্রোপার্টিতে যেয়ে ContextMenuStrip এর ডানে (none) শব্দের উপর ক্লিক করলেই দেখতে পাবেন ContextMenuStrip1। এটি চয়েজ করুন।

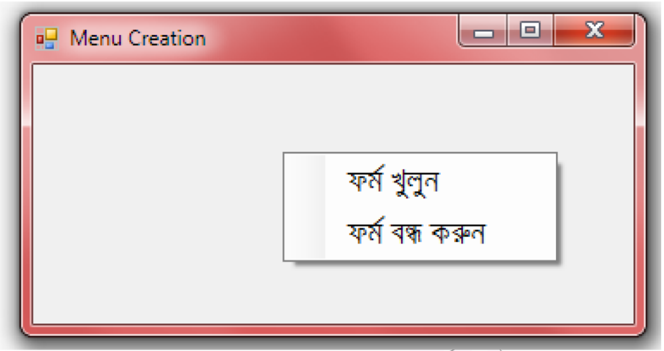


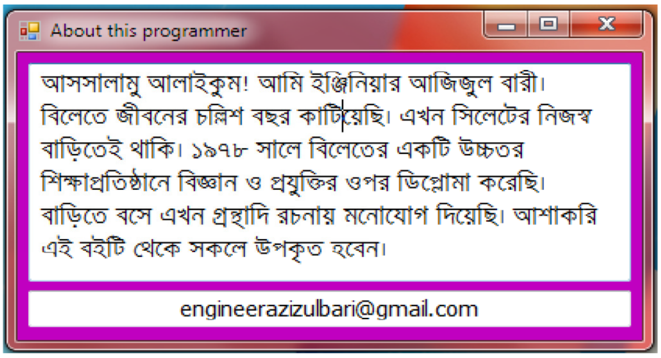

```
Private Sub ফর্মখুলুন_Click(sender As Object, e As EventArgs) Handles ফর্মখুলুন.Click
    If Form2.Visible = False Then
        Form2.Show()
        End If
    End Sub
Private Sub ফর্মবন্ধকরুন_Click(sender As Object, e As EventArgs) Handles ফর্মবন্ধকরুন.Click
    If Form2.Visible = True Then
        Form2.Close()
    End If
End Sub
```

প্রোগ্রাম রান করুন: পরের পৃষ্ঠার প্রথম [বামের] চিত্রে রানকালীন অবস্থায় প্রথম ফর্মে রাইট-ক্লিক করার পর কনটেক্স মেনু আসতে দেখাচ্ছে। ডানের চিত্রে কনটেক্স মেনু আইটেম থেকে ফর্ম খুলার লাইনটি চয়েজ করায় ফর্মটি খুলেছে।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

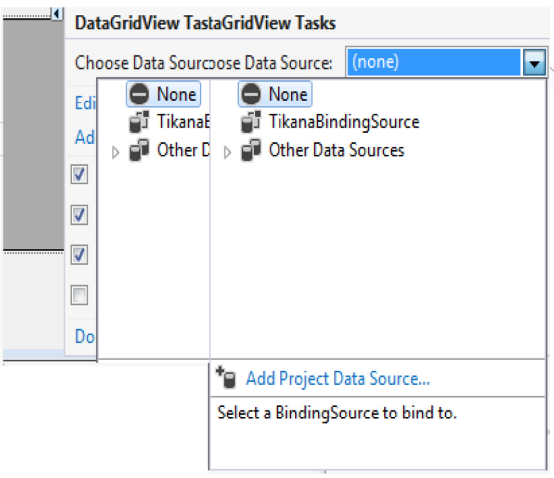
উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

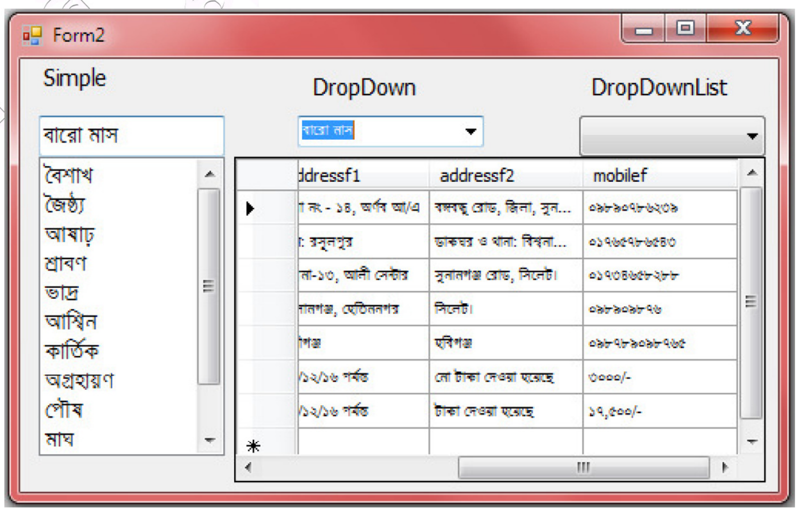




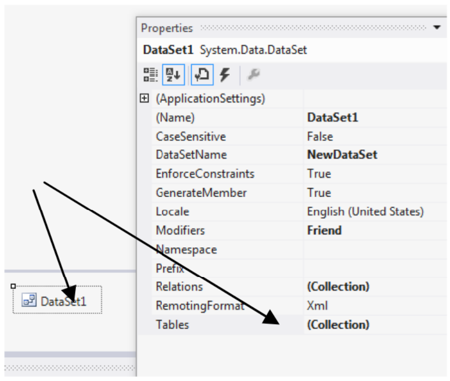
11. DataGridView: এটা ডাটাবেইজ প্রদর্শনের গ্রিড যা দেখতে অনেকটাই এক্সেস ডাটাবেইজের টেবিলের মতো। একটি উদাহরণ দেখে নিই।

মনে আছে কি- আমাদের ঠিকানা ডাটাবেইজের কথা? গ্রন্থের ১৪০ পৃষ্ঠায় চিত্রসহ নির্দেশনা দেওয়া আছে কিভাবে ডাটাবেইজের সঙ্গে যোগাযোগ সৃষ্টি করা যায়। এখানে অতিরিক্ত যা তাহলো, ডাটাগ্রিডভিউ ফর্মে নিয়ে আসার সাথে সাথেই একটি জানালা আসবে (নিচের বায়ের চিত্র)। যে কম্বোতে লিখা Add Project Data Source... আছে সেখানে যেয়ে নতুন ডাটাবেইজ আনার জন্য ক্লিক করুন। এরপর ১৪০ পৃষ্ঠায় যেভাবে নির্দেশিত হয়েছে সেভাবে ডাটাবেইজের সঙ্গে সংযোগ সৃষ্টি করে প্রোগ্রাম রান করুন [নিচের ডানের বায়ের চিত্র দ্রঃ]।





12. DataSet: এটা হলো ডাটাবেইজ 'মেমোরির মধ্যে উপস্থাপন'। এতে থাকতে পারে ডাটাটেবিল, ডাটারো ও ডাটাকলাম ইত্যাদি। ডাটাসেট ফর্মে নিয়ে আসলে তা নিচের ধূসর এলাকায় দেখা যাবে। এর প্রোপার্টি জানালায় যেয়ে বিভিন্ন ধরনের সম্পর্ক সৃষ্টি করতে পারেন। যেমন আপনি যদি চান ডাটাসেট মেমোরিতে একটি এক্সেস টেবিল রাখবেন, তাহলে আপনি এর প্রোপার্টিতে যেয়ে Tables -এ ক্লিক করলে ডানে কালেকশন শব্দ ও একটি বোতাম দেখতে পাবেন। এতে ক্লিক করলেই টেবিলের সঙ্গে যোগাযোগ তৈরি করতে পারেন।

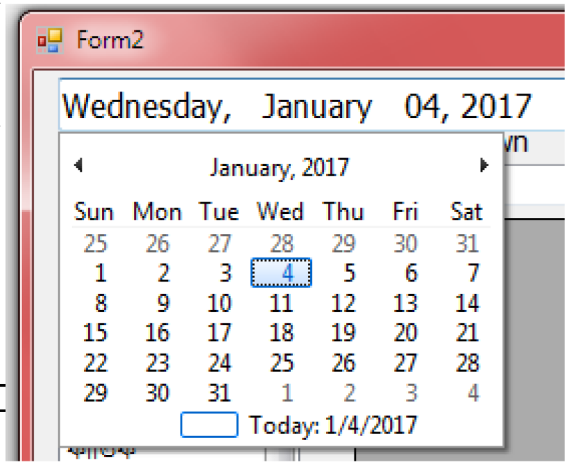


13. DateTimePicker: এই কন্ট্রোল দিয়ে তারিখ ও সময় প্রদর্শন করা যায়। কন্ট্রোলের প্রোপার্টিতে যেয়ে যেবাবে চান সেভাবে প্রদর্শনও সেট করতে পারেন। পরের পৃষ্ঠায় একটি চিত্রে আমরা এর ব্যবহার দেখিয়েছি। একটি ফর্মে এই কন্ট্রোল যুক্ত করে তার বিভিন্ন প্রোপার্টি কিভাবে কাজ করে তা দেখে নিতে পারেন।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

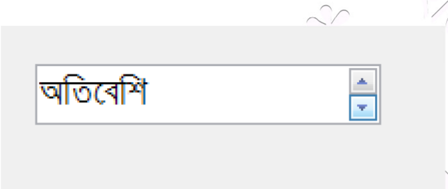
ডানের চিত্রে ডেইটটাইমপিকার দ্বারা ‘আজকের তারিখ’ একটি ক্যালেন্ডারে কিভাবে দেখানো হয় তা তুলে ধরেছি।



14. DirectoryEntry: এটার কাজ হলো ডাইরেক্টরির ভেতর কি হচ্ছে তা জানা। কোডের মাধ্যমে একে ব্যবহার করে যায়।

15. DirectorySearcher: এটার কাজ হলো ডমেইন সার্ভিস হাইআর্কিতে অনুসন্ধান চালানো।

16. DomainUpDown: এটা অনেকটা **CheckedListBox** এর মতো কাজ করে। এই কন্ট্রোলের ভেতর একটি লিস্ট থাকতে পারে। ডানের আপ-ডাউন এ্যারোতে ক্লিক করে করে লিস্ট দেখা যায়। নিচের চিত্রটি দেখুন।

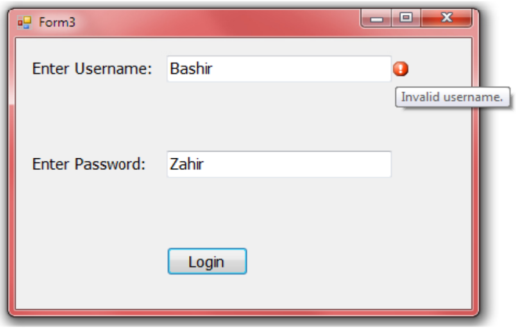


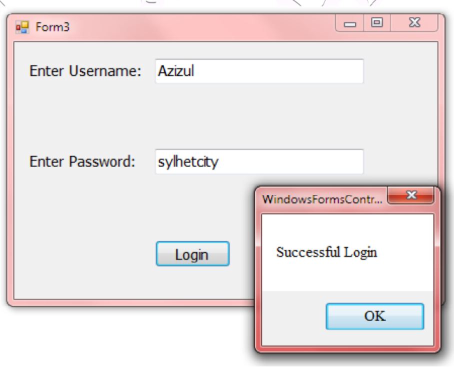
অবশ্য কোডের মাধ্যমে জানা ও সেট করা যায় লিস্টের কোন্ আইটেম চয়েজ করা হয়েছে বা হবে। যেমন:

DomainUpDown1.SelectedIndex = 0

17. ErrorProvider: এটা একটি সুবিধা। অর্থাৎ কোনো আইটেম ব্যবহারকারী প্রবেশ করলেন, আর তা আপনি যেভাবে চান তা হয় নি। এটা জানানোর জন্য আইকন ও মেসেজ ব্যবহারকারীকে দেখাতে পারেন। একটি উদাহরণ দ্বারা ব্যাপারটি স্পষ্ট হবে।

মনে করুন আপনি একটি লগিন ফর্ম বানাবেন। ব্যবহারকারী দু’টি ফিল্ডে ইউজারনেইম ও পাসওয়ার্ড প্রবেশ করবেন। আপনি একটি **ErrorProvider** দ্বারা লগিনের ভেলিডিটি চেক করতে পারেন। দেখুন নিচের চিত্র দু’টো ও পরের কোডগুলো।






```
Private Sub btLogin_Click(sender As Object, e As EventArgs) Handles btLogin.Click
    If TextBox1.Text <> "Azizul" Then
        ErrorProvider1.SetError(TextBox1, "Invalid username.")
        Exit Sub
    Else
        ErrorProvider1.SetError(TextBox1, "")
    End If
    If TextBox2.Text <> "sylhetcity" Then
        ErrorProvider1.SetError(TextBox2, "Invalid password.")
        Exit Sub
    Else
        ErrorProvider1.SetError(TextBox2, "")
    End If
    MsgBox("Successful Login")
End Sub
```

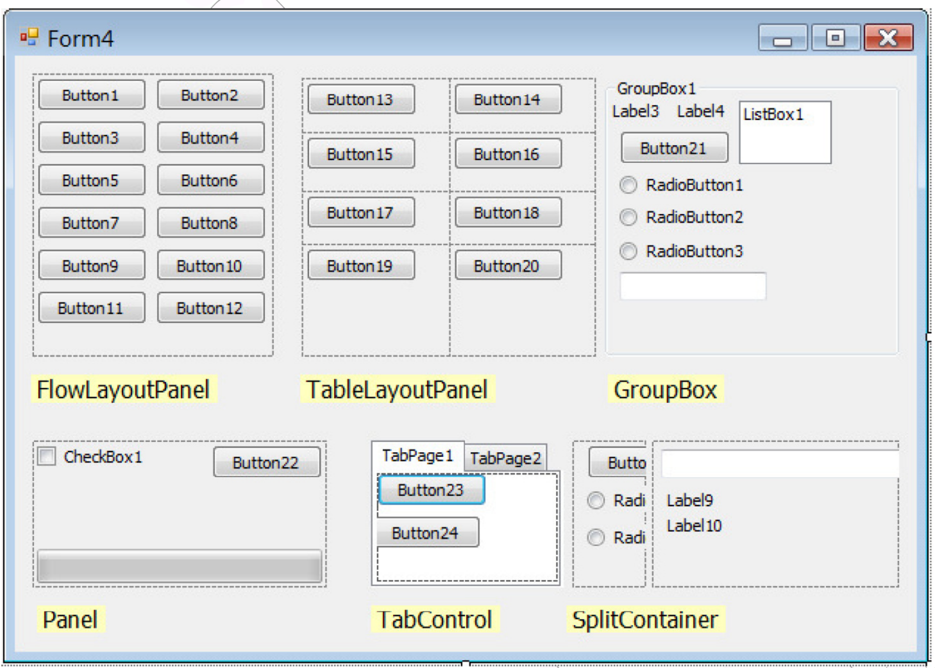
১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

18. EventLog: এটার মাধ্যমে কোনো এ্যাপলিকেশনের ইভেন্ট লগ [ঘটনাপ্রবাহ রেকর্ড] তৈরি করা যায়। এটা আমরা খুব অল্প ক্ষেত্রেই ব্যবহার করে থাকি। সুতরাং এ সম্পর্কে বেশি বলার নেই।

19. FileSystemWatcher: এটার কাজ হলো ফাইল সিস্টেমে কী রদবদল হচ্ছে তা জানা ও প্রয়োজনে রিপোর্ট করা। এটা একটি এডভান্স স্তরের ক্রিয়া তাই এটা ব্যবহার পদ্ধতি এখানে দেখানোর প্রয়োজন নেই।

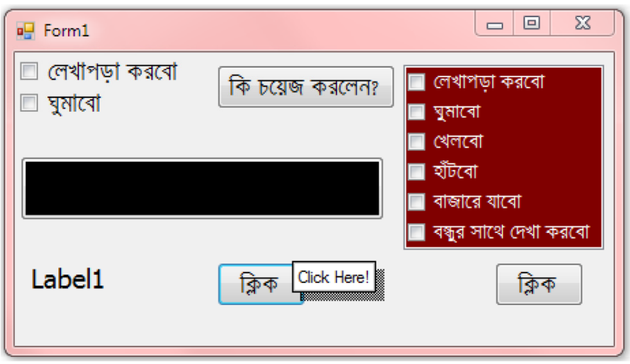
20. FlowLayoutPanel: -এটা Panel জাতীয় কয়েকটি কন্ট্রোলের একটি। অন্যান্যগুলো হলো: TableLayoutPanel, GroupBox, TabControl, SplitContainer এবং Panel। এগুলো মূলত ডিজাইনকালে সুবিধার জন্য ব্যবহৃত হয়। নিচের চিত্রটি দেখুন।



22. HelpProvider: যে কোন কন্ট্রোলের জন্য সাহায্য নিশ্চিত করার উপায় হিসাবে এই কন্ট্রোলটি ব্যবহৃত হয়ে থাকে। যেমন মনে করুন, আপনি চাচ্ছেন একটি বোতামের একশন কি হবে তা ব্যবহারকারীকে জানাবেন। আপনি চাইলে টুলটিপ ব্যবহার করতে পারেন। তবে এরচেয়েও উন্নতমানের পথ হলো হেল্পপ্রভাইডার। এটির মাধ্যমে শুধুমাত্র সামান্য কিছু লেখা নয়- আরো একটি সংরক্ষিত হেল্প ফাইলও দেখাতে পারেন। ব্যবহারকারী কন্ট্রোলটি ফোকাসে আসার পর কিবোর্ডের ‘’ -এ প্রেস করলেই হেল্পপ্রভাইডার সক্রিয় হয়ে ওঠবে। আসুন, একটি উদাহরণ দেখে নিই- বিষয়টি এতে স্পষ্ট হয়ে যাবে। আমরা ফর্মের লোড ইভেন্ট সাবে এই কোডটুকু লিখবো:

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
   HelpProvider1.SetHelpString(Button3, "Click Here!")
End Sub
```

অবশ্য ফর্মের গ্রে এলাকায় একটি HelpProvider1 আইটেম তো আনতে হবে। টটুলবাক্সে যেয়ে এই আইটেমে ডবলক্লিক করলেই কাজটি সারবে। রান করার পর বোতামটি ফোকাসে এলে ‘F1’ কী চাপুন। ডানের চিত্র দেখুন।



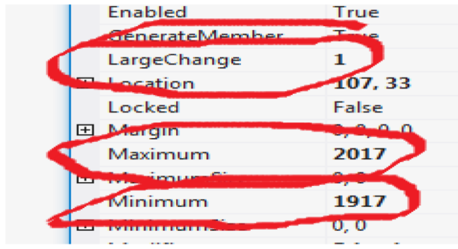
১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

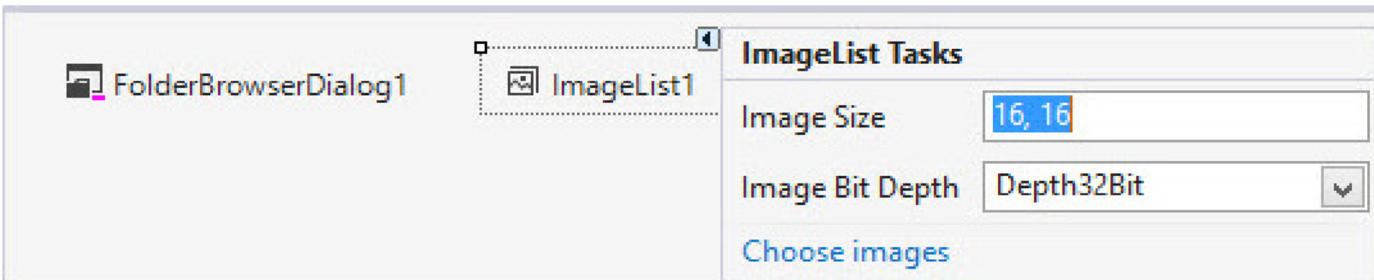
23. HScrollBar: এটা ও ‘**VScrollBar**’ কন্ট্রোলের কাজ হলো যথাক্রমে ব্যবহারকারী একটি অংশে ধরে অনুভূমিক ও উপর-নিচে স্ক্রল করতে পারবেন। আমরা একটি উদাহরণ দ্বারা এগুলো বুঝিয়ে দিচ্ছি।



বায়ের চিত্রে আমরা দু’টি স্ক্রল আইটেম দেখছি। এগুলোর মাধ্যমে লেবেল দু’টোর সংখ্যা রদবদল করা যায়। আমরা ডিজাইন টাইমে সর্বোচ্চ ও সর্বনিম্ন সংখ্যা নির্ধারণ করে দিতে পারি। কোন্ কোন্ প্রপার্টি পরিবর্তন করে তা সম্ভব নিচের চিত্রে দেখিয়েছি।



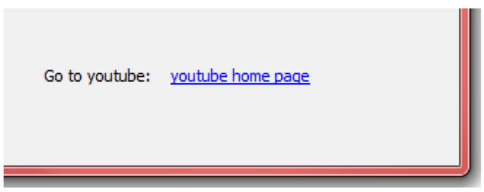
24. ImageList: এটা ছবির একটি লিস্ট স্টোর করে যাতেকরে অন্যান্য কন্ট্রোল বা কোড এগুলো কাজে লাগাতে পারে। আপনি প্রথমে ফর্মে একটি ইমেজলিস্ট নিয়ে আসবেন টুলবক্স থেকে। দেখবেন, এর ডানে একটি ছোট্ট এ্যারো আছে। সিলেক্ট করলেই ইমেজলিস্ট ট্যাস্ক্‌স লিস্ট আসবে। আপনি সেখানে ইমেজ সম্পর্কে তথ্যাদি লিখে রাখতে পারেন। ইমেজ লিস্ট নিয়েও আসতে পারেন। দেখুন নিচের চিত্রটি।



25. Label: এটা সম্পর্কে আর কী বলবো। আমরা তো সব সময় এটা ব্যবহার করে আসছি। ব্যবহারকারীকে দৃশ্যত কিছু তথ্য প্রদানের জন্য এটি কাজে লাগানো হয়। অবশ্য রানটাইমে কোডোর মাধ্যমে এর টেক্সট, ভিজিবল ইত্যাদি প্রোপার্টি পরিবর্তনও করা যায়।

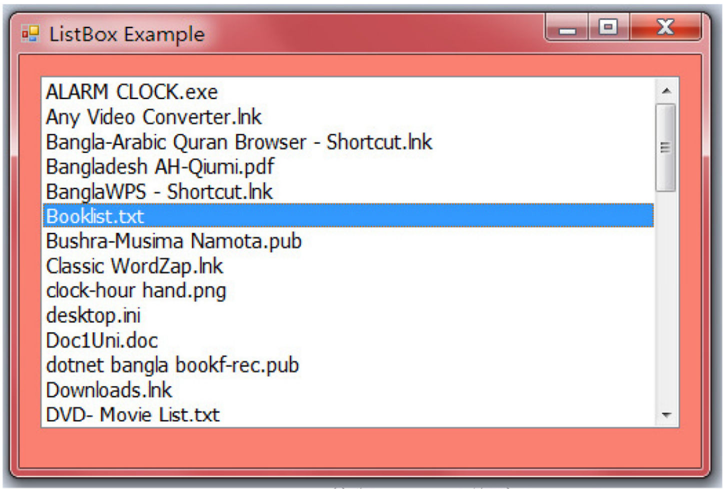
26. LinkLabel: এটাও মূলত একটি লেবেল। তবে ওয়েভ পেইজে ন্যাভিগেশনের জন্য এটি সহজে ব্যবহার করা যায়। বা ফর্মের ভেতরেও অন্যান্য কাজের জন্য এটি ব্যবহারযেগ্য। এর লেখা থাকে নীল রঙের। এছাড়া আন্ডারলাইনও থাকে। এর উপরে মাউস গেলে হাত আইকনে পরিণত হয়। দেখুন নীচের চিত্রটি। লিংকলেবেলের বেশ কয়েকটি প্রোপার্টি আছে। এগুলো রদবদল করে দেখতে পারেন। একটি ওয়েভসাইটে সরাসরি ভিজিটের জন্য ফর্মের লিংকলেবেলে নিচের কোডটুকু লিখুন।

```
Private Sub LinkLabel1_LinkClicked(sender As Object, e As LinkLabelLinkClickedEventArgs) Handles
LinkLabel1.LinkClicked
        ' open the website URL set to the Tag property
        ' (www.youtube.com)
        Process.Start(LinkLabel1.Tag.ToString)
End Sub
```



১৭তম পরিচ্ছেদ: *Windows Form Controls*

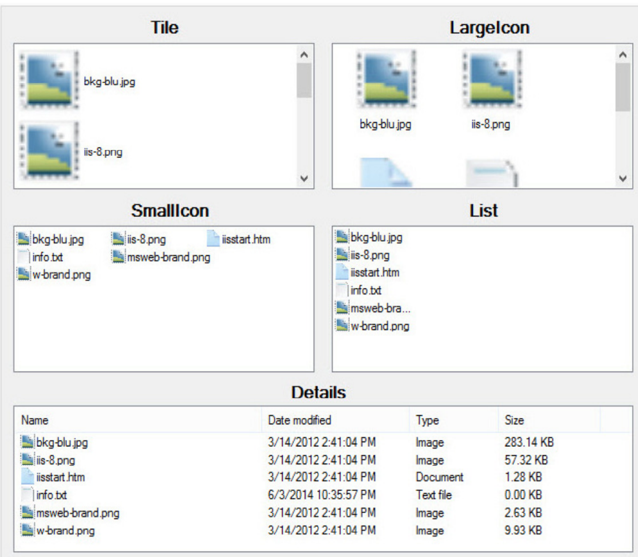
উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*



27. ListBox: এটাও আমরা ইতোমধ্যে ব্যবহার করেছি। এতে থাকে নিয়ন্ত্রণযোগ্য কিছু লিস্ট আইটেম। লিস্ট হবে ০ ইনডেক্স। অর্থাৎ চারটি আইটেম থাকলে ইনডেক্স হবে ০,১,২ ও ৩। ইনডেক্স দ্বারা, কিংবা লাইনের সিলেকশন টেক্সট দ্বারা লিস্ট বাক্সের আইটেম সনাক্ত করা যায়। আমরা একটি উদাহরণ দেখাবো। বায়ের চিত্রে একটি জানালার মধ্যে আমরা ফাইলের একটি লিস্ট দেখছি। বাস্তবে এই জানালায় একটি লিস্ট বাক্স আছে। কোডিং এর মাধ্যমে আমরা কম্পিউটারের ডেস্কটপ ফল্ডারের সকল ফাইলের নাম এই বাক্সে প্রদর্শন করেছি। নিচে এই কোডটুকু দেখুন। একটি ফর্ম আপনার প্রজেক্টে যুক্ত করুন। এতে টুলবাক্স থেকে নিয়ে আসুন একটি লিস্টবক্স। ফর্মের কোথাও ডবল ক্লিক করুন। কোড এডিটরের Form_Load ইভেন্ট সাবে নিচের কোডটি লিখুন। আপনি চাইলে অন্য যে কোনো ফল্ডারের নামও দিতে পারেন। বরং বিভিন্ন ফল্ডানের নাম দিয়ে টেস্ট করে দেখুন।

```
Private Sub Form6_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Dim mydoc As String = My.Computer.FileSystem.SpecialDirectories.Desktop
    Dim path As String = IO.Path.Combine(mydoc)
    Dim dir As New IO.DirectoryInfo(path)
    ' populate the listbox control
    For Each file In dir.GetFiles
        ' avoid jpg images
        If Not file.Extension = ".jpg" Then
            ListBox1.Items.Add(file.Name)
        End If
        Next
End Sub
```

28. ListView: এটাতে ফাইলের লিস্ট দেখানো যায়। তবে আরো বেশি তথ্যসহ। মোট ৫ উপায়ে লিস্টভিউ কন্ট্রোলে ফাইল নাম প্রদর্শন সম্ভব। বায়ের চিত্রটি দেখুন।



ক. Tiles: আইটেমের নামের বামপাশে বড়ো বাড়ো আইকন দেখায়।

খ. LargeIcon: আইটেমের নামের উপরে বড়ো আইকন দেখায়।

গ. SmallIcon: নামের বায়ে ছোট্ট আইকন প্রদর্শন করে।

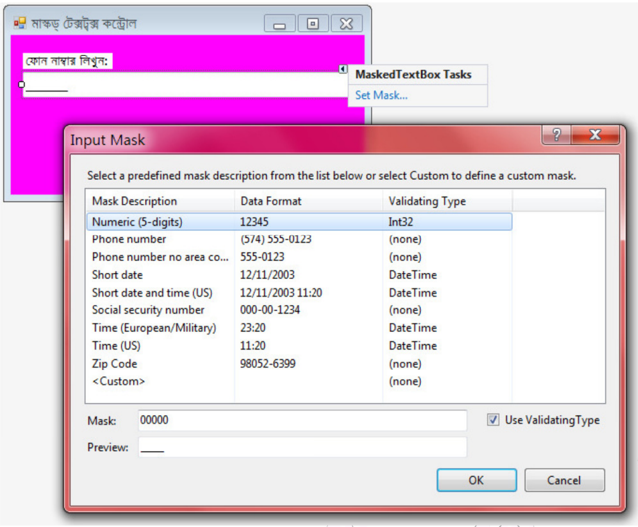
ঘ. List: নামের বায়ে ছোট্ট আইকন দেখায় ও প্রতিটি আইটেম একেকটি লাইনে থাকে।

ঙ. Details: ছোট্ট আইকনসহ ফাইলের পূর্ণাঙ্গ তথ্য প্রদর্শন করে।

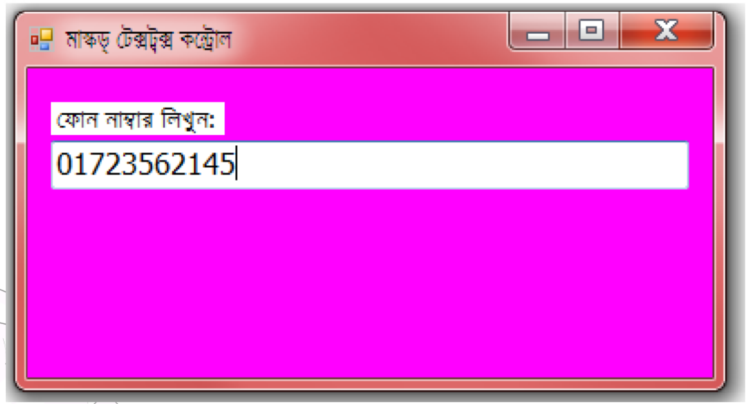
29. MaskedTextBox: এটা টেক্সট বাক্সের মতোই কাজ করে। তবে অতিরিক্ত যা তাহলো ব্যবহারকারী ফিল্ডের মধ্যে কী লিখবেন তা নির্ধারণ করা যায়। এক ধরনের ভেলিডিটি ফিল্ড এটি। উদাহরণ দেখুন।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

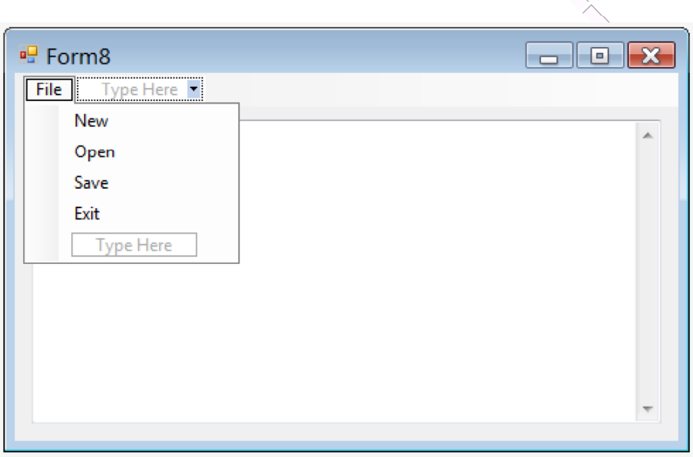


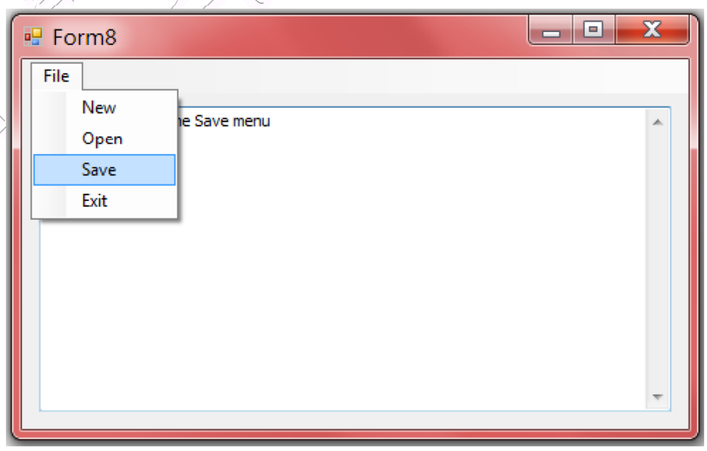
টুল বাক্স থেকে ফর্মে যুক্ত করুন **MaskedTextBox** একটি । টেক্সবাক্সের ডানের দিকে উপরের কোণে যে এ্যারো দেখা যাবে তাতে ক্লিক করুন। চয়েজ করুন Set Mask …। বায়ে চিত্রিত জানালাটি আসবে। এখান থেকে যে কোনো একটি চয়েজ করে 'OK ক্লিক করুন। এবার প্রোগ্রাম রান করুন। দেখবেন আপনার



চয়েজকৃত উপায়েই লেখা ঢুকানো সম্ভব হচ্ছে টেক্সট ফিল্ডে। লক্ষ করুন, ফিল্ডের মধ্যে ১১টি মাত্র সংখ্যা ঢুকানো সম্ভব হয়েছে।

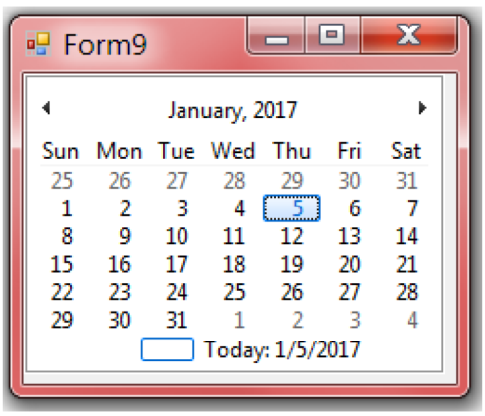
30. MenuStrip: এ কন্ট্রোলের মাধ্যমে ফর্মের মধ্যে মেনু, সাবমেনু, মেনু আইটেম ইত্যাদি তৈরি করা যায়। প্রত্যেকটির সঙ্গে কোড যুক্ত করে কাঙ্ক্ষিত একশনও আঞ্জাম দেওয়া যায়। নিচের চিত্রগুলো দেখুন।





31. MessageQueue এটা একটি উচ্চ পর্যায়ের ক্লাস। সার্ভারের সাথে মেসেজ যোগাযোগ কিউ নিয়ন্ত্রণে এটি ব্যবহৃত হয়। এ টুলবক্স আইটেম সম্পর্কে আর অতিরিক্ত আলোচনা গ্রন্থের স্কোপের বাইরে। সুতরাং আর কিছু বলার নেই। এ সম্পর্কে অতিরিক্ত জানতে চাইলে ইন্টারনেটের এই লিংক দেখতে পারেন: http://msdn.microsoft.com/en-us/library/system.messaging.messagequeue.aspx

32. MonthCalender বুঝতেই পারছেন, এটা হলো পুরো মাসের ক্যালেন্ডার কন্ট্রোল। ক্যালেন্ডারের রেঞ্জ বিরাট: ১৭৫৩ থেকে ৯৯৯৮ সাল পর্যন্ত এ রেঞ্জের ব্যাপ্তি। আপনি এরই ভেতর যে কোনে তারিখ যথাক্রমে সর্বনিম্ন ও সর্বোচ্চ হিসেবে প্রোপার্টিতে যেয়ে সেট করতে পারেন। ডানের চিত্রটি দেখুন।

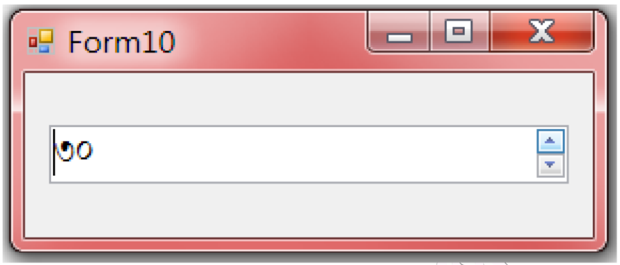


33. NotifyIcon: এটা প্রোগ্রাম রানকালে অদৃশ্য থাকে। একে ব্যবহার করা

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

যায় সিস্টেম ট্রেতে আইকোন প্রদর্শন করে।



34. **NumericUpDown:** এ কন্ট্রোল একটি সংখ্যা প্রদর্শন করে। এতে থাকে একটি আপ-ডাউন এ্যারো। আপনি প্রোপার্টিতে যেয়ে সর্বোচ্চ ও সর্বনিম্ন সংখ্যা, কত করে বাড়বে বা কমমে ক'টি ডেসিমেলে স্থান হবে ইত্যাদি সেট করতে পারেন। কন্ট্রোলের ValueChanged ইভেন্ট সক্রিয় হয় যখনই ব্যবহারকারী কিংবা কোড এর মান পরিবর্তন করে। এই ইভেন্ট সাবকে ধরে কোডিং করা যায়। দেখুন বায়ের চিত্রটি।

35. **OpenFileDialog** -এটার মাধ্যমে ফাইল খোলার ডায়লগ প্রদর্শিত হয়। এটা একটি শক্তিশালী কন্ট্রোল। অবশ্য ভিজুয়্যাল বেসিক.নেট বেশ ক'টি অনুরূপ ডায়লগ সৃষ্টি করে দিয়েছে। অপেন ডায়লগের কাজ হলো কম্পিউটার থেকে কোনো ফাইল সিলেক্ট করে খোলার জন্য প্রস্তুত করা। ডায়লগের ফলাফল হয় '**OK**' না হয় '**Cancel**'। এই কন্ট্রোলে বেশ ক'টি প্রোপার্টি আছে। এগুলো রানটাইমে ব্যবহার করা যায়। যেমন Filter প্রোপার্টির মাধ্যমে নিশ্চিত করতে পারেন যে ব্যবহারকারী শুধুমাত্র .jpg ও .png ফাইল টাইপ খোলা ছাড়া অন্য কিছু দেখতেই পারেন না। আমরা কোডে লিখতে পারি:

OpenFileDialog1.Filter = "Jpeg Files (*.jpg)|*.jpg|Png Files (*.png)|*.png"

যে ফাইল সিলেক্ট করা হয়েছে তার নাম জানতে হলে আমরা ব্যবহার করতে পারি FileName প্রোপার্টি, যেমন:

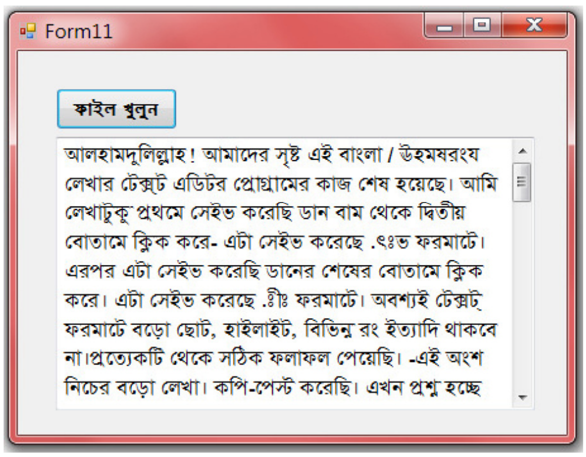
```
Private Sub BrowseButton_Click(sender As Object, e As EventArgs) Handles BrowseButton.Click
    If OpenFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
        SelectedFile.Text = OpenFileDialog1.FileName
    End If
End Sub
```

আরো একটি উল্লেখযোগ্য প্রোপার্টি হচ্ছে MultiSelect । এটা দ্বারা সিদ্ধান্ত নেওয়া যায় ব্যবহারকরী একাধিক ফাইল সিলেক্ট করতে পারবে কি না। যেমন:

```
Private Sub BrowseButton_Click(sender As Object, e As EventArgs) Handles BrowseButton.Click
    If OpenFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
        For Each file As String In OpenFileDialog1.FileNames
            MessageBox.Show(file)
        Next
    End If
End Sub
```

মেসেজ বাক্সে একে একে সকল ফাইলের নাম প্রদর্শিত হবে। এবার আমরা ছোট্ট একটি উদাহরণ দেখে নেবো।

একটি নতুন ফর্ম প্রজেক্ট তৈরি করুন। যোগ করুন একটি বোতাম, একটি টেক্সট বাক্স ও একটি অপেন ফাইল ডায়লগ কন্ট্রোল। বোতামে ডবল ক্লিক করুন। কোড জানালাতে ফর্মের ক্লাসের ভেতর যে ইভেন্ট সাব আসবে তার ভেতর লিখুন:



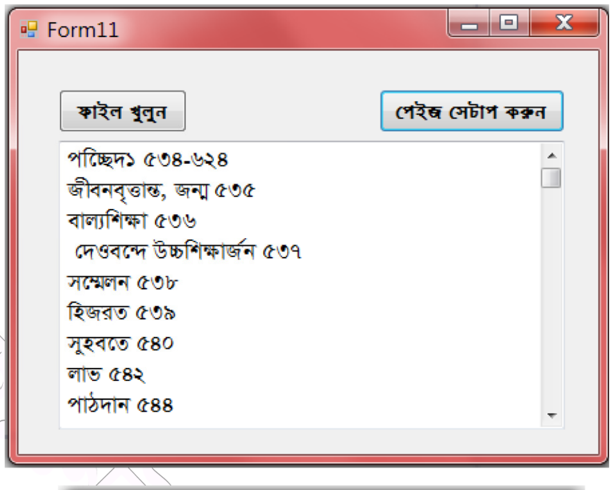

```
Public Class Form11
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button1.Click
        ' Displays an OpenFileDialog so the user can select a Cursor.
        Dim openFileDialog1 As New OpenFileDialog()
```

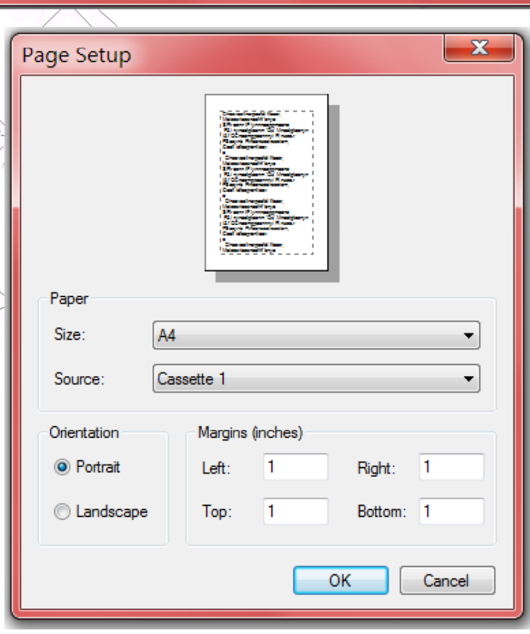
১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
 openFileDialog1.Filter = "Text Files|*.txt"
   openFileDialog1.Title = "Select a Text File"
   If openFileDialog1.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
      Dim sr As New System.IO.StreamReader _(openFileDialog1.FileName)
      TextBox1.Text = (sr.ReadToEnd)
      sr.Close()
   End If
 End Sub
End Class
```

36. PageSetupDialog: এই ডায়লগটি দ্বারা পেইজ সেটাপ করা যায়। একটি উদাহরণ দেখুন। আমরা ইতোমধ্যে অপেনডায়লগ নিয়ে যে প্রোগ্রামটি করেছিলাম সেটি আবার খুলুন। ফর্মে যোগ করুন একটি পেইজসেটাপডায়লগ ও একটি 'পেইজ সেটাপ করুন' বোতাম। এ বোতামের ইভেন্ট সাবরুটিনে লিখুন:






```
Private Sub Button2_Click(sender As Object, e As EventArgs)
Handles Button2.Click
    ' Initialize the dialog's PrinterSettings property to hold user
    ' defined printer settings.
    PageSetupDialog1.PageSettings = _
       New System.Drawing.Printing.PageSettings
    ' Initialize dialog's PrinterSettings property to hold user
    ' set printer settings.
    PageSetupDialog1.PrinterSettings = _
       New System.Drawing.Printing.PrinterSettings
    'Do not show the network in the printer dialog.
    PageSetupDialog1.ShowNetwork = False
    'Show the dialog storing the result.
    Dim result As DialogResult = PageSetupDialog1.ShowDialog()
    ' If the result is OK, display selected settings in
    ' ListBox1. These values can be used when printing the
    ' document.
    If (result = DialogResult.OK) Then
       Dim results() As Object = New Object() _
       {PageSetupDialog1.PageSettings.Margins, _
       PageSetupDialog1.PageSettings.PaperSize, _
       PageSetupDialog1.PageSettings.Landscape, _
       PageSetupDialog1.PrinterSettings.PrinterName, _
       PageSetupDialog1.PrinterSettings.PrintRange}
    End If
End Sub
```

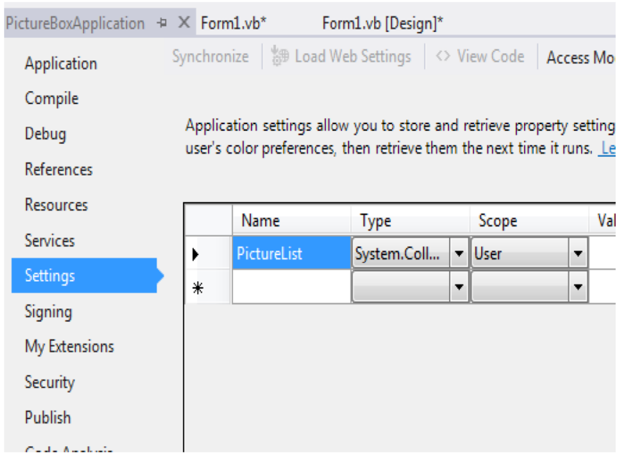
37. Panel: প্যানেল কন্ট্রোল মূলত গ্রুপ বাক্সের মতো। আমরা ১৭৭ পৃষ্ঠায় এর ব্যবহার দেখেছি। সুতরাং এখানে আর পুনরাবৃত্তি করছি না।

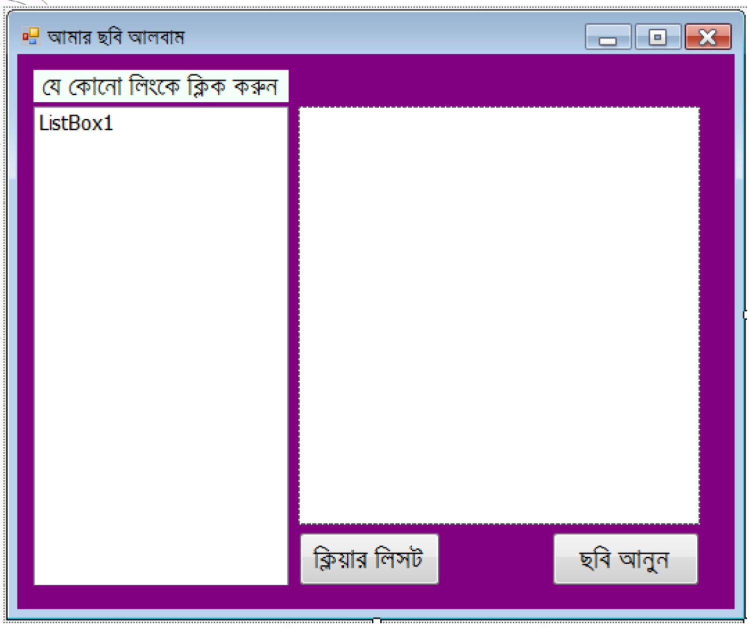
38. PerformanceCounter: এটা হলো উইন্ডোজ এনটি [Windows NT] পারফরম্যান্স কাউন্টার কম্পোনেন্ট। এটা একটি উচ্চ পর্যায়ের কন্ট্রোল। কম্পিউটারের সিস্টেমের সঙ্গে এর ইন্টারেকশন। বর্তমান গ্রন্থে এর ওপর আদৌ কোনো অতিরিক্ত আলোচনা ঠিক হবে না।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

39. PictureBox: এই কন্ট্রোলের মাধ্যমে ছবি প্রদর্শন করা যায়। এছাড়া এটি একটি গ্রাফিক্স অবজেক্ট হিসেবেও কাজ করে যার মধ্যে আপনি বিভিন্ন ধরনের আকার রানটাইমে অঙ্কন করতে পারেন। ডিজাইন টাইমে দু'টি প্রোপার্টি খুব ব্যবহারোপযোগী। এগুলো হলো তার ইমেজ [Image] এবং সাইজমোড [SizeMode] প্রোপার্টি। প্রথমটি দ্বারা যে কোনো ছবি ডিফোল্ট হিসাবে প্রদর্শন ও দ্বিতীয়টি দ্বারা ছাবিটি কিভাবে প্রদর্শিত হবে তা নির্দিষ্ট করা যায়। ছবিটি কন্ট্রোলের পুরো এলাকা জুড়ে, মধ্যখানে, বড় হয়ে ইত্যাদি কোন্ উপায়ে প্রদর্শিত হবে- তা জিাইনটাইমে করার সুযোগ সৃষ্টি করে দিয়েছে দ্বিতীয় প্রোপার্টিটি। পিকচার বাক্স সম্পর্কে আরো কিছু জানতে আমরা একটি সুন্দর এ্যাপলিকেশন তৈরি করবো। এই এ্যাপলিকেশন হবে ছবি আলবামের মতো। আপনি যতোটা প্রয়োজন ছবি কম্পিউটারের যে কোনো ফল্ডার থেকে এনে পিকচার বাক্সে প্রদর্শন করতে পারেন। আরো যা পারবেন তাহলো পরবর্তিতে প্রোগ্রাম রান করলে সকল ছবির লিংক একটি লিস্ট বাক্সে দেখতে পাবেন। শুধু মাত্র যে কোনো লিংকে ক্লিক করলেই ছবিটি ভেসে আসবে পিকচার বাক্সে। সুতরাং আপনার ছবি আলবাম বানাতে নিচের পদক্ষেপগুলো অনুসরণ করুন।





১. একটি ফর্ম প্রজেক্ট তৈরি করুন। এর নামকরণ করুন: MyPictureAlbum।

২. ফর্মে টুলবাক্স থেকে নিয়ে আসুন দু'টি বোতাম, একটি পিকচার বাক্স, একটি লেবেল, একটি লিস্ট বাক্স ও একটি অপেন ফাইল ডায়লগ। এরপর ইচ্ছেমতো ডিজাইন করুন। দেখুন আমার ডিজাইনটি।

৩. Solution Explorer এ যেয়ে ডবল ক্লিক করুন, My Project। একটি জানালা আসবে। নিচে এর একটি অংশ দেখানো হলো।

৪. সিলেক্ট করুন, Settings। Name কলামে টাইপ করুন: PictureList। পাশের Type কলামের এ্যারোতে যেয়ে সিলেক্ট করুন: System.Collections.Specialized.StringCollection। ব্যস! এটি বন্ধ করুন। এই সেটিংয়ের কারণে আমরা লিস্ট বাক্সের আইটেমগুলো সংরক্ষণ করতে পারবো।

৫. এবার কিছু কোডিং লাগবে। (ক) 'ছবি আনুন' বোতামে ডবল ক্লিক করে কোড এডিটর খুলুন। বোতামের ক্লিক ইভেন্টে লিখুন:

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim result As DialogResult
    'set filter
    OpenFileDialog1.Filter = "JPEG Files (*.jpg)|*.jpg|Bitmap Files(*.bmp)|*.bmp| Gif files(*.gif)|*.gif| Png files
(*.png)|*.png"
    'Enable multi-file names selection
    OpenFileDialog1.Multiselect = True
    'Open file dialog
    result = OpenFileDialog1.ShowDialog
    If result = DialogResult.OK Then
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
        For Each filename In OpenFileDialog1.FileNames
            'add selected file names to listbox
            ListBox1.Items.Add(filename)
        Next
    End If
End Sub
```

এই কোডের কাজ হলো OpenFileDialog1 এর মাধ্যমে এক বা একাধিক ছবির পূর্ণাঙ্গ প্যাথ স্ট্রিং লিস্টবাক্সে নিয়ে আসা। প্রথম লাইন দ্বারা আমরা ডায়লগ থেকে যে ফলাফল আসবে তার জন্য একটি ডিক্লারেশন করেছি। দ্বিতীয় লাইন দ্বারা নির্দিষ্ট করেছি কোন্ কোন্ ধরনের ইমেজ ফাইল আনা যাবে। এর পরের লাইনে অপেন ডাইলগের Multiselect নামক প্রোপার্টি সত্য হিসাবে সেট করেছি। এর ফলে একই সঙ্গে একাধিক ছবি সিলেক্ট করতে পারবেন ব্যবহারাকারী। এর পরের লাইনে ব্যবহারকারী ক্যানসেল না করলে আমরা প্রতিটি সিলেক্টেট ছবির ফাইন নাম [তাদের প্যাথ নেইম] লিস্ট বাক্সে নিয়ে এসেছি। অবশ্য ইতোমধ্যে লিস্ট বাক্সে কিছু থাকলে তা মুছে যাবে না।

৬. লিস্ট বাক্সে ডবল ক্লিক করুন। যে সাব আসবে তাতে লিখুন এই কোডটুকু:

```
Private Sub ListBox1_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
    ListBox1.SelectedIndexChanged
    'View image in picturebox when a file name was selected from the list
    PictureBox1.Image = Image.FromFile(ListBox1.SelectedItem.ToString)
End Sub
```

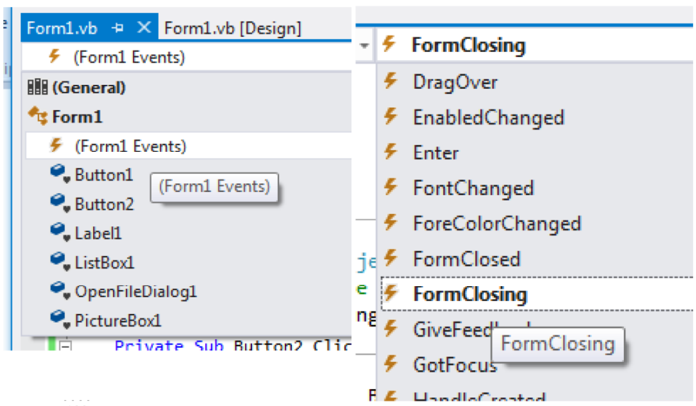
এখানো আমরা একটি মাত্র লাইন লিখেছি। লিস্ট বাক্সে যে ইনডেক্সে ক্লিক করা আছে সে প্যাথ নাম অনুযায়ী পিকচার বাক্সে ছবি এনে প্রদর্শনের নির্দেশ দেওয়া হয়েছে এ লাইনের মাধ্যমে।

৭. এবার ডবল ক্লিক করুন ক্লিয়ার লিস্ট বোতামে। ক্লিক ইভেন্ট সাবে লিখুন:

```
'clear list button code
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
     PictureBox1.Image = Nothing
     Dim Res As Integer = MsgBox("সকল প্যাথ নাম মুছে দেবেন?", MsgBoxStyle.YesNo)
     ' MsgBox(Res)
     If Res = 6 Then
        ListBox1.Items.Clear()
     End If
End Sub
```

এই সাবের প্রথম লাইন দ্বারা প্রদর্শিত ইমেজ (পিকচার বাক্সে থাকলে) মুছে দিয়েছি। এরপর একটি ওয়ার্নিং দ্বারা ব্যবহাকরীকে জানানো হচ্ছে সত্যিই কি লিস্টটি মুছে দেবেন? ফলাফল সত্যি হলে [৬] পরের লাইন নির্দেশ দেওয়া হয়েছে লিস্ট বাক্সের সবকিছু মুছে ফেলে দিতে।

৮. Form Load এবং Form Closing ইভেন্ট সাবের কোড: আমরা এ দু'টো ইভেন্ট ধরে কিছু কাজ করেছি। লোড হওয়ার সময় আমরা চাই সিস্টেম সেটিংয়ে সংরক্ষিত লিস্টটি লিস্ট বাক্সে নিয়ে আসতে। আর বন্ধ হওয়ার সময় আমরা চাই লিস্ট বাক্সের আইটেমগুলো আবার সিস্টেম সেটিংয়ে সংরক্ষণ করতে- কারণ, ইতোমধ্যে ব্যবহারকারী লিস্টে পরিবর্তন আনতে পারেন। প্রথম সাবরুটিন আসবে ফর্মের টাইলেট বারে ডবল ক্লিক করলে। কিন্তু দ্বিতীয়টি ফর্মের ইভেন্ট লিস্ট থেকে আনতে হবে। দেখুন ডানের চিত্রটি। এটা আনার ক্লিক পরম্পরা: ক্লিক Form1.vb > Form1 > (Form1 Event > FormClosing। ব্যস! `Form1_FormClosing` নামক একটি ইভেন্ট হ্যান্ডলার আসবে।



১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

যাক, প্রথমে আমরা ফর্ম লোডের কোডের ওপর ব্যাখ্যা দেবো। নিচে এই কোডগুলো দেখানো হয়েছে।

```
'get whatever in settings to the ListBox
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles Me.Load
        For Each item As String In My.Settings.PictureList
            ListBox1.Items.Add(item) 'my.settings items adding to listBox1
            Next
    End Sub
```

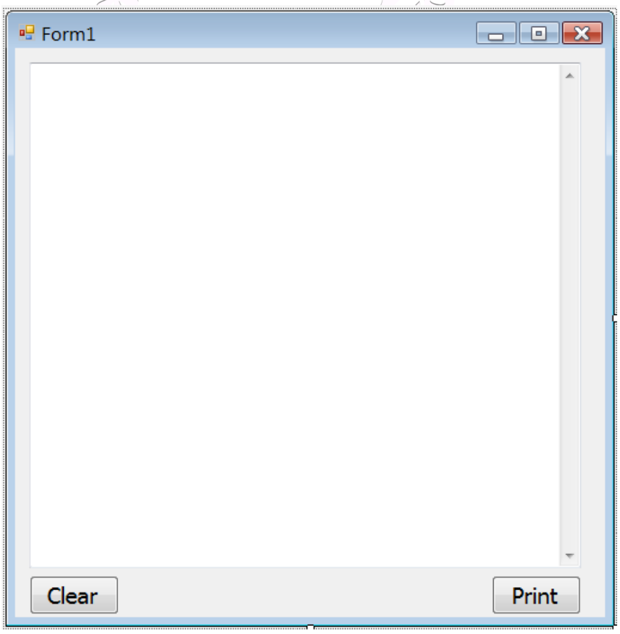
লক্ষ করুন, পিকচার লিস্টে যে ক'পি রেকর্ড আছে তা আমরা একে একে লিস্ট বাক্সে এনে প্রদর্শন করেছি। সুতরাং আলবামের সকল সংরক্ষিত ছবি পিকচার বাক্সে আনতে কোনো বাধা নেই। শুধুমাত্র লিংকে ক্লিক করলেই কাজ হবে। এবার দেখে নেবো ফর্ম ক্লোজিং সাবের কোড।

```
'save whatever in the list to setting as lines of strings
Private Sub Form1_FormClosing(sender As Object, e As FormClosingEventArgs) Handles Me.FormClosing
        My.Settings.PictureList = New Specialized.StringCollection() 'pictureList collection Instant declaration
            For Each item As String In ListBox1.Items
                My.Settings.PictureList.Add(item) 'ListBox items add to settings one by one
                My.Settings.Save() 'and save it
            Next
End Sub
```



এ কোডগুলোর প্রথম লাইন সেটিংয়ে সংরক্ষিত লিস্ট মুছে দেবে ও একটি নতুন কালেকশন তৈরি করবে। এরপর লিস্ট বাক্সের প্রতিটি লাইন একে একে ঐ কালেকশনে নিয়ে সংরক্ষণ করবে। সুতরাং আলবামের সকল লিংক সংরক্ষিত হবে। ব্যবহারকারী যখন প্রোগ্রাম আবার রান করবেন তখন ঐ কালেকশন থেকে লিংকগুলো উপরের ফর্ম লোড ইভেন্ট সাবের কোডের মাধ্যমে লিস্ট বাক্সে চলে আসবে। সুতরাং আমরা সত্যিকার অর্থে একটি ছবি আলবাম তৈরি করলাম। বায়ে এর রানটাইমকালীন একটি চিত্র দিয়েছি। দেখে নিন।

40. PrintDialog: এই ডায়লগ ও PrintDocument-অবজেক্ট একত্রে কাজ করে আপনার টেক্সট প্রিন্ট করতে পারেন ডিফোল্ট প্রিন্টারে। আমরা একটি উদাহরণ তুলে ধরছি।



একটি নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে যুক্ত করুন দু'টি বোতাম ও একটি টেক্সট বাক্স। এরপর টুলবাক্স থেকে ডবল ক্লিক করে সলুশনে নিয়ে আসুন **PrintDialog** এবং PrintDocument অবজেক্ট। ডানের চিত্রটি দেখুন।

এবার ক্লিয়ার বোতামে ডবল ক্লিক করুন এবং সাবের ভেতর নিচের কোড লাইনটি লিখুন।

```
Private Sub Button2_Click(sender As Object, e As EventArgs)
Handles Button2.Click
        TextBox1.Text = ""
End Sub
```

এই কোড টেস্টট্ বাক্সকে খালি করে দেবে। এবার ডবল ক্লিক করুন

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

নিচের ধূসর এলাকায় পড়ে থাকা আমাদের PrintDocument1 -কে। একটি হ্যান্ডলার সাব আসবে ফর্মের কোড জানালায়। এতে নিচের কোডটুকু লিখুন।

```
'set the document font and colour
Private Sub PrintDocument1_PrintPage(sender As Object, e As Printing.PrintPageEventArgs) Handles
PrintDocument1.PrintPage
    Dim font1 As New Font("TonnyBanglaMJ", 14, FontStyle.Regular)
    e.Graphics.DrawString(TextBox1.Text, font1, Brushes.Black, 100, 100)
End Subr
```

এই কোডের কাজ হলো টেক্সট বাক্সের লেখাটুকুর ফন্ট, সাইজ, ফন্ট স্টাইল, ব্রাশ ইত্যাদি সেট করা। এবার প্রিন্ট বোতামে ডবল ক্লিক করুন। নিচের কোডটুকু লিখুন ইভেন্ট হ্যান্ডলারে।

```
'show setup dialog and then print using PrintDocument Object
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
   If TextBox1.Text = "" Then
      MsgBox("Please write something!")
      Exit Sub
    End If
    Dim result As DialogResult = PrintDialog1.ShowDialog()
    If result = 1 Then
      PrintDocument1.Print()
    End If
End Sub
```

এই কোডের কাজ হলো প্রথমে নিশ্চিত করা টেক্সট বাক্সে কিছু লেখা আছে। এরপর প্রিন্ট ডায়লগ দেখানো সেটিংয়ের জন্য। তারপর প্রিন্ট ডকুমেন্ট অবজেক্টকে এ সেটিং মুতাবিক প্রিন্ট করার জন্য বলা। ব্যস! এটুকুই। লক্ষ করুন, মাঝখানে ডিম স্টেটমেন্ট এর মাধ্যমে আমরা প্রিন্ট ডায়লগ প্রদর্শন করেছি। আমরা ব্যবহারকারীর একশন কি জানতে চাই। কারণ তিনি প্রিন্ট না করে ক্যানসেল বোতামেও ক্লিক করতে পারেন। প্রিন্ট বোতামের ফলাফল ১ এবং ক্যানসেল বোতামের ফলাফল ২। সুতরাং রেজাল্ট ১ হলেই প্রিন্টের নির্দেশ দিতে পারি। আপনার টেক্সট বাক্সে যাই থাকুক- এ ক্ষেত্রে বিজয় কিবোর্ডে বাংলায় লিখা থাকবে, তা-ই ডিফোল্ট প্রিন্টারে প্রিন্ট হবে। আমি টেস্ট করে দেখেছি। আপনিও দেখে নিন।

41. PrintDocument: আমরা এইমাত্র এ অবজেক্টের কার্যকারিতার ওপর আলোচনা করেছি উদাহরণসহ। সুতরাং এখানে আর কিছু বলার নেই।

42. PrintPreviewControl: প্রিন্টিংয়ের জন্য প্রিন্ট ডায়লগ ও প্রিন্ট ডকুমেন্টই যথেষ্ট। তবে সময় সময় আমরা পিন্ট্রের প্রিভিউ দেখতেও চাই। এই কম্পোনেন্টের কাজ এটাই।

43. PrintPreviewDialog: আমাদের সৃষ্ট উদাহরণটিকে কিছুটা পরিবর্তন করে নিলে এই ডায়লগের কার্যকারিতাও দেখতে পারি। এটার কাজও প্রিন্ট করা, তবে প্রথমে প্রদর্শিত হবে একটি প্রিভিউ। প্রিন্টিংসহ বেশ ক'টি ফাংশন আছে। আসুন, ইতোমধ্যে আলোচিত প্রজেক্টকে কিছুটা মডিফাই করি।

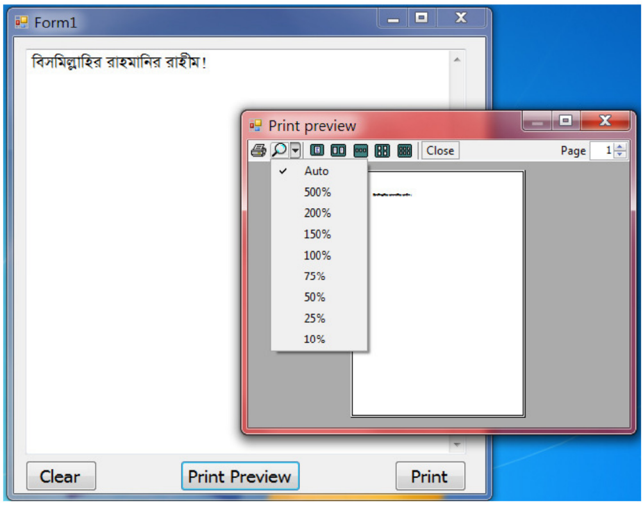
প্রথমেই আমরা টুলবাক্সে যেয়ে ফর্মে আরো একটি বোতাম ও **PrintPreviewDialog** যুক্ত করবো। বোতামের নামকরণ করবো: Print Preview। নতুন বোতামে ডবল ক্লিক করে সাবরুটিনে নিচের কোডটুকু লিখুন।

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    If TextBox1.Text = "" Then
      MsgBox("Write something!")
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
    Else
      PrintPreviewDialog1.Show()
    End If
End Sub
```

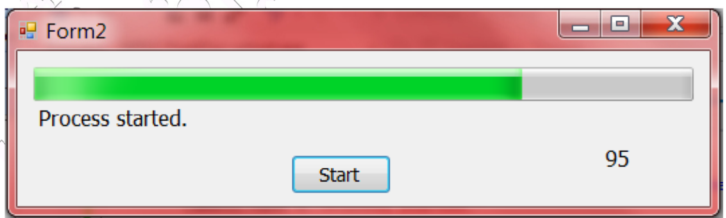


এই যা- আর কিছু লাগবে না। প্রিন্ট প্রিভিউ করেও এবার প্রিন্টিং করতে পারেন। ডানের চিত্রটি দেখুন।

44. Process: এটার কাজ হলো কম্পিউটারে চলন্ত প্রসেসগুলো সম্পর্কে তথ্য সংগ্রহ করা ও প্রয়োজনে প্রদর্শন করা। এটা মূলত সিস্টেম ডায়াগনস্টিক্স এর একটি ফাংশন। এডভান্স স্তরের প্রোগ্রামাররা একে ব্যবহার করতে পারেন। সুতরাং আমাদের ক্ষেত্রে এর ওপর আলোচনা আদৌ সঠিক হবে না।

45. ProgressBar: ওয়েভ ব্রাউজার উদহরণ প্রোগ্রাম তৈরির সময় আমরা এর ব্যবহার দেখেছি। কোনো দীর্ঘ প্রসেসের সময় ব্যবহারকারীকে ফিডব্যাক দিতে এটি কাজে লাগানো যায়। এর দু'টি গুরুত্বপূর্ণ প্রোপার্টি আছে: Minimum এবং Maximum। প্রথমাবস্থায় বারটি সম্পূর্ণ রংহীন থাকবে আর দ্বিতীয় অবস্থায় সম্পূর্ণরূপে রঙে পরিপূর্ণ হবে। একটি উদাহরণ দেখে নিই।

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Label1.Text = "Process started."
    For x As Integer = 0 To 100
      Label2.Text = x
      ProgressBar1.PerformStep()
      wait(1) 'wait for a millisecond
    Next
    ProgressBar1.Refresh()
    Label1.Text = "Process Ended."
End Sub
```




```
' Loops for a specificied period of time (milliseconds)
Private Sub wait(ByVal interval As Integer)
    Dim sw As New Stopwatch
    sw.Start()
    Do While sw.ElapsedMilliseconds < interval
      ' Allows UI to remain responsive
      Application.DoEvents()
    Loop
    sw.Stop()
End Sub
```

উপরের wait সাবরুটিন দ্বারা নির্দিষ্ট মিলিসেকেন্ড পরিমাণ অপেক্ষার ব্যবস্থা করেছি। এটা প্রগ্রেস বারের মাত্রা বৃদ্ধির পরই বার বার কল করেছি যাতেকরে বারের গতিটি দৃশ্যমান হয়।

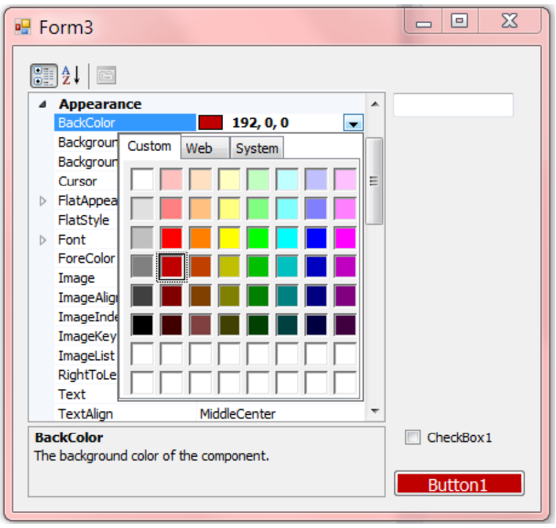
46. PropertyGrid: এটা আইডিই [Integrated Development Environment] এর মতো বস্তুর প্রোপার্টিগুলো দেখায়। কন্ট্রোলের দু'টি গুরুত্বপূর্ণ প্রোপার্টি হলো: ১. SelectedObject এবং ২. SelectedObjects। প্রথমটির অর্থ একটি অবজেক্ট ও দ্বিতীয়টির অর্থ একাধিক অবজেক্ট। রানটাইমে অবজেক্টের প্রোপার্টি রদবদলও করা যায়। আমরা একটি উদাহরণ দেখাবো।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

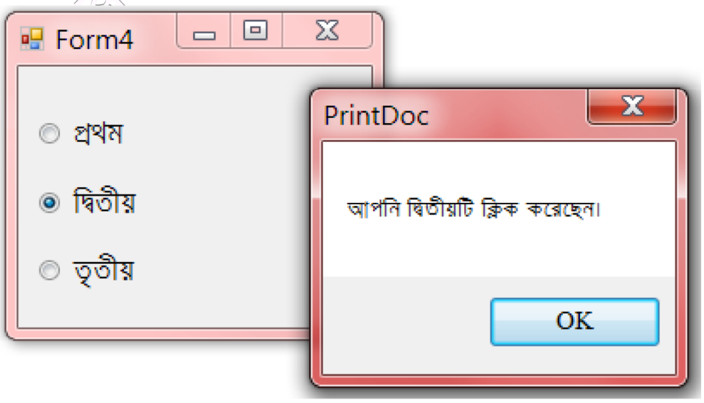
একটি উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করে ফর্মের ভেতর একটি বোতাম, একটি চেক বাক্স, একটি টেক্সট বাক্স ও একটি প্রোপার্টি গ্রিড যুক্ত করুন। এরপর নিচের কোডগুলো বস্তুর ক্লিক ইভেন্ট সাবে লিখুন।

```
'make SelectedObject as Button1
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    PropertyGrid1.SelectedObject = Button1
End Sub
'make SelectedObject as CheckBox1
Private Sub CheckBox1_CheckedChanged(sender As Object, e As
EventArgs) Handles CheckBox1.CheckedChanged
    PropertyGrid1.SelectedObject = CheckBox1
End Sub
'make SelectedObject as TextBox1
Private Sub TextBox1_MouseEnter(sender As Object, e As EventArgs)
Handles TextBox1.MouseEnter
    PropertyGrid1.SelectedObject = TextBox1
End Sub
```



আমরা এসব কোড দ্বারা প্রোপার্টি গ্রিডের সিলেক্টেড অবজেক্ট প্রোপার্টি বস্তুর সাথে সম্পৃক্ত করেছি মাত্র। দেখুন ফলাফল।

47. RadioButton: এ কন্ট্রোলে থাকে একটি গোল বোমাত ও পাশে কিছু লেখা। ব্যবহারকারী গোল বোতামে ক্লিক করলে কেন্দ্রে রঙিন একটি বৃত্ত আসে। কয়েকটি বোতাম একই সঙ্গে রাখা যায়। কোনটি ক্লিক করা আছে তা কোডের মাধ্যমে জেনে প্রয়োজনীয় ক্রিয়া সম্পাদন করাই হলো এর মূল কার্যকারিতা। দেখুন নীচের চিত্রটি।



48. RichTextBox: আমরা ইতোমধ্যে এই কন্ট্রোল নিয়ে কাজ করেছি। এটা টেক্সট বাক্সের মতোই- তবে কিছু মৌলিক পার্থক্য আছে। যেমন, এই বাক্সে যে কোনো ফন্ট [ইংরেজি, বাংলাসহ] ব্যবহার করা যায়। টেক্সট ফরমাটিংও সম্ভব। দেখুন আমাদের সৃষ্ট ‘বাংলা টেক্সট এডিটর’ [পৃষ্ঠা ১৩৬]।

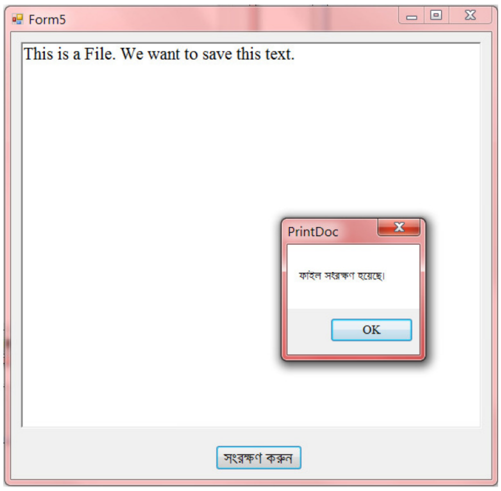
49. SaveFileDialog: এটা দ্বারা একটি সংরক্ষণ ডায়লগ দেখানো যায়। এটিও আমরা উক্ত ‘বাংলা টেক্সট এডিটর প্রোগ্রামে ব্যবহার করেছি। তবে এখানে একটি ছোট্ট উদাহরণ দিচ্ছি। মনে করুন একটি জানালায় আছে রিচটেক্সবক্স কন্ট্রোল। আপনি এতে কিছু লিখে ফাইলে সংরক্ষণ করবেন। এটা আঞ্জাম দিতে প্রথমেই একটি ইউজার ইন্টারফেইস তৈরি করুন। দেখুন নিচের রানটাইমের চিত্রটি। এতে থাকবে একটি রিচটেক্সবক্স, একটি ‘সংরক্ষণ করুন’ বোতাম ও একটি সেইভফাইলডায়লগ। এবার দেখুন প্রোগ্রামের কোডগুলো।

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    If RichTextBox1.Text = "" Then
      Exit Sub
    End If
    SaveFileDialog1.DefaultExt = "*.rtf"
    SaveFileDialog1.Filter = "RTF Files|*.rtf"
    SaveFileDialog1.CreatePrompt = True
    If SaveFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
      RichTextBox1.SaveFile(SaveFileDialog1.FileName, RichTextBoxStreamType.RichText)
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
        MsgBox("ফাইল সংরক্ষণ হয়েছে।")
        End If
    End Sub
```

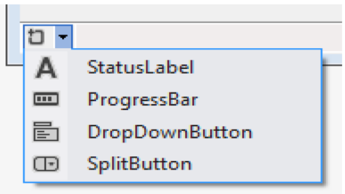


50. SerialPort: -এটা দ্বারা কম্পিউটারের যে কোনো ফিজিক্যাল সিরিয়েল পোর্টের বিভিন্ন প্রোপার্টি সম্পর্কে অবগত হওয়া যায়। এগুলোর মধ্যে আছে: baud rate, break signal, Data Set Ready (DSR) state, port name, parity, and stop bits ইত্যাদি। নোট করুন, এসব ব্যাপার অনেকটা এডভান্স স্টেজের প্রোগ্রামিংয়ের সঙ্গে জড়িত। সুতরাং এ গ্রন্থে এসব ব্যাপার নিয়ে আলোচনা আদৌ সঠিক হবে না।

51. ServiceController: এটাও উচ্চ পর্যায়ের প্রোগ্রামারদের সুবিধার্তে তৈরি করেছে মাইক্রসফট। উইন্ডোজ সার্ভিস প্রসেস ও মেথড এতে জড়িত। আমরা এ সম্পর্কে অতিরিক্ত কিছু বলতে পারবো না।

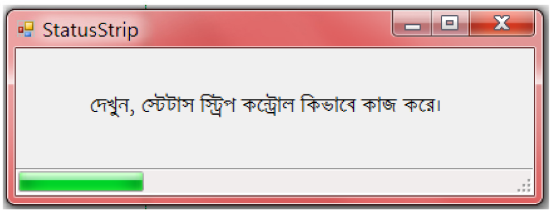
52. SplitContainer: এ কন্ট্রোল সম্পর্কে কিছুটা ব্যাখ্যা ১৭৭ পৃষ্ঠায় হয়েছে। এটি মূলত ডিজাইন কম্পোটেন্ট। এ সম্পর্কে এখানে আর কিছু বলছি না।

53. Splitter: এটার দ্বারাও ডিজাইনকালে কিছু কাজ করা যায়।



54. StatusStrip: এটা অনেকটা টুলস্ট্রিপের মতো। ফর্মের নিচে এটি প্রদর্শিত হয়। এর উদ্দেশ্য হলো স্টেটাস তথ্যাদি সংক্ষিপ্তভাবে দেখানো। এতে কয়েকটি বস্তু থাকে: স্টেটাস লেবেল, প্রগ্রেস বার, ড্রপডাউনবাটন এবং স্প্লিটবাটন। এগুলোর প্রতিটি আবার ফর্মের একেকটি কন্ট্রোল দ্বারা রেপ্রেজেন্ট করে। যেমন: প্রগ্রেসবারকে রেপ্রেজেন্ট করে টুলস্ট্রিপবার কন্ট্রোল। উদাহরণ থেকে ব্যাপারটি স্পষ্ট হবে। একটি ফর্মে যুক্ত করুন একটি স্টেটাস স্ট্রিপ কন্ট্রোল। ছোট্ট এ্যারোতে যেয়ে প্রথমে চয়েজ করুন: ProgressBar। ফর্মে ডবলক্লিক করে এর কোড এডিটর খুলুন। যুক্ত করুন নিচের কোডগুলো লোড ইভেন্ট সাবে:

```
While ToolStripProgressBar1.Value < 100
    ToolStripProgressBar1.Value += 5
End While
```



রান করলে বায়ের চিত্রটি দেখতে পাবেন।

55. TabControl: এই কন্ট্রোল পৃষ্ঠায় পৃষ্ঠায় গ্রুপকৃত তথ্যাদি প্রদর্শন করে থাকে। আমরা এর ব্যবহার ১৭৭ পৃষ্ঠায় দেখেছি। সুতরাং এখানে আর অতিরিক্ত কিছু বলার নেই- তবে! একটি উদাহরণ দেখে নিই।

একটি নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে একটি ট্যাব কন্ট্রোল যুক্ত করুন। আমাদের উদাহরণে মোট ৫টি ট্যাব আছে। কন্ট্রোলের ডানের উপরের কোণে যে ছোট্ট এ্যারো আছে তাতে ক্লিক করে Abb Tab দ্বারা যতোটি প্রয়োজন ট্যাব যুক্ত করতে পারেন। প্রত্যেক ট্যাবের আলাদা জানালা আছে যাতে যুক্ত করা যায় অন্যান্য কন্ট্রোল। আমি প্রথম ট্যাবে একটি পিকচার বাক্স ও একটি লেবেল যুক্ত করেছি। পিকচার বাক্সে দেখাবো একটি ছবি এবং লেবেলে এ ছবি সম্পর্কে তথ্য থাকবে। অন্যান্য ট্যাবেও বিভিন্ন বস্তু যুক্ত করেছি। এবার দেখে নিন কোডটুকু।

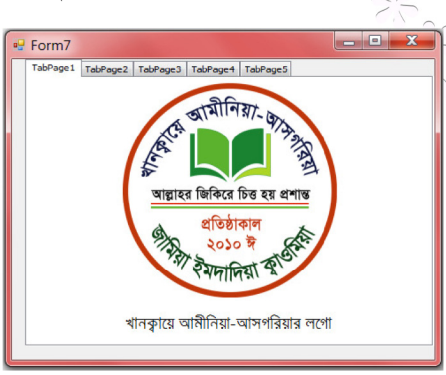
১৭তম পরিচ্ছেদ: *Windows Form Controls*

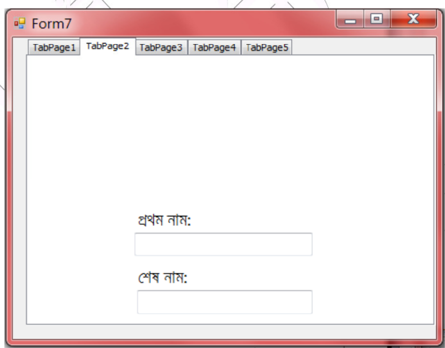
উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

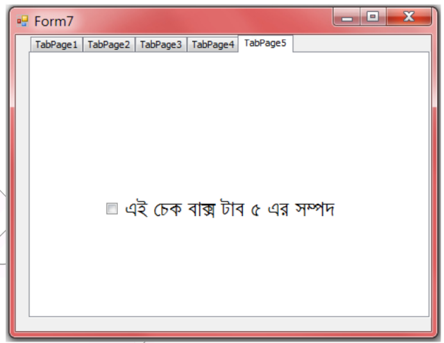
```
Private Sub TabControl_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      ' select the tabpage #3
      TabControl1.SelectedIndex = 4
End Sub

Private Sub TabControl1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
TabControl1.SelectedIndexChanged
      If TabControl1.SelectedIndex = 0 Then
         PictureBox1.Image = My.Resources.LogoA
      End If
End Sub
```

উপরের প্রথম সাবরুটিনে আমরা ট্যাবের ডিফোল্ট ইনডেক্স কি হবে তা নির্দেশ করেছি। আর দ্বিতীয় সাব দ্বারা ইনডেক্স সিলেকশন ০ হলে ছবি প্রদর্শনের কথা বলেছি। ছবিটি অবশ্য My.Resources- এ ইতোমধ্যে এনে রেখেছিলাম। এবার দেখুন রানটাইমের ক'টি চিত্র।







56. TableLayoutPanel: এটা ডিজাইনকালে ব্যবহারের একটি কন্ট্রোল। এটার ওপর আলোচনা ইতোমধ্যে পৃষ্ঠা ১৭৭-এ হয়ে গেছে। সুতরাং এখানে আর কিছু বলার নেই।

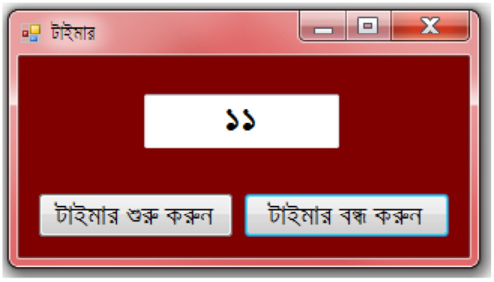
57. TextBox: এ কন্ট্রোলটি সম্ভবত সর্বাধিক বেশি ব্যবহৃত হয় ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিংয়ে। পুরো এ বইটি জুড়ে টেক্সট বাক্স ব্যবহারের উদাহরণ আছে। এখানে অতিরিক্ত যা বলার তাহলো, এটা রিচটেক্সটবক্স থেকে অনেকটা সহজ, এর ভেতর যে কোনো একটিমাত্র ফন্ট ব্যবহার করা যায়, ফন্টের রঙ একই হতে হবে। বিশেষ তেমন কোনো ফর্মাটিংও সম্ভব নয়। মোটকথা বেশি সুবিধার জন্য রিচ টেক্সটবক্স কন্ট্রোল অনেকটা ব্যবহারোপযোগী। টেক্সবক্স কন্ট্রোলের গুরুত্বপূর্ণ ক'টি প্রোপার্টি হলো: MultiLine, MaxLength, ReadOnly, WordWrap, ScrollBars and Text ।

58. Timer: এটার কাজ হলো কোনো ফাংশন বা প্রসিডোরে ইন্টারভ্যল বা বিরতি দেওয়া। টাইমার কন্ট্রোল সলুশনের ধূসর এলাকায় পড়ে থাকে। একে কাজে লাগাতে হলে তার বিভিন্ন মেথড ও ফাংশন কল করতে হয় কোডে। একটি উদাহরণ দেখে নিই। নতুন একটি ফর্ম প্রজেক্ট তৈরি করুন। একটি টাইমার কন্ট্রোল প্রজেক্টে নিয়ে আসুন টুলবাক্স থেকে। আরো যুক্ত করুন দু'টি বোতাম ও একটি টেক্সট বাক্স। প্রথম বোতামের টেক্সট ভ্যালু হবে, 'টাইমার শুরু করুন' এবং দ্বিতীয়টির ভ্যালু হবে, 'টাইমার বন্ধ করুন'। টেক্সট বাক্সের ফন্ট সাইজ কিছুটা বাড়িয়ে দিন। এর ভ্যালু প্রপার্টি লিখুন ১। আমার মনে হয় উভয় বোতামের ফন্ট সাইজও কিছুটা বাড়িয়ে দিন যাতে স্পষ্ট দেখা যায়। টাইমারের প্রোপার্টিতে যেয়ে Interval প্রোপার্টির ভ্যালু লিখুন ১০০০। এর অর্থ ১০০০ মিলিসেকেন্ড বা ১ সেকেন্ড। এবার প্রথমে ডবল ক্লিক করুন টাইমার কন্ট্রোলে। কোড এডিটর জানালা আসবে। সেখানে Timer1_Tick ইভেন্ট সাবরুটিন। এ সাবের ভেতর লিখুন:

```
Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
    TextBox1.Text = Val(TextBox1.Text) + 1
End Sub
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*



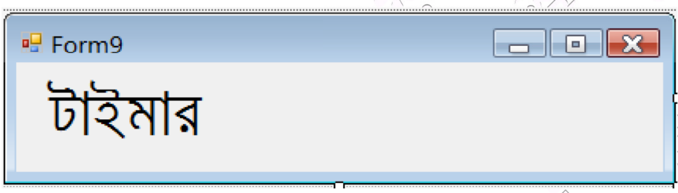
এরপর 'টাইমার শুরু করুন' বোতামে ডবল ক্লিক করে তার ইভেন্ট সাবে লিখুন:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    Timer1.Start()
End Sub
```

সবশেষে 'টাইমার বন্ধ করুন' বোতামে ডবল ক্লিক করে তার ইভেন্ট সাবে লিখুন:

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
    Timer1.Stop()
End Sub
```

প্রোগ্রাম শুরু করে উভয় বোতামে ক্লিক করে দেখুন কিভাবে টেক্সট বাক্সে সংখ্যাটি বাড়ে ও বন্ধ হয়। আমরা এই টাইমার কন্ট্রোল দ্বারা একটি ডিজিটাল ঘড়ি বানাতে পারি। দেখুন এটা বানানো কতো সহজ!



নতুন প্রজেক্ট তৈরি করুন। ফর্মে একটি লেবেল নিয়ে আসুন। এর ফন্ট সাইজ ইত্যাদি পরিবর্তন করুন। যেভাবে বায়ের চিত্রে দেখানো হয়েছে। এর ফন্ট হবে বাংলা। আমরা বাংলা ডিজিটাল ঘড়ি বানাবো, কেমন?

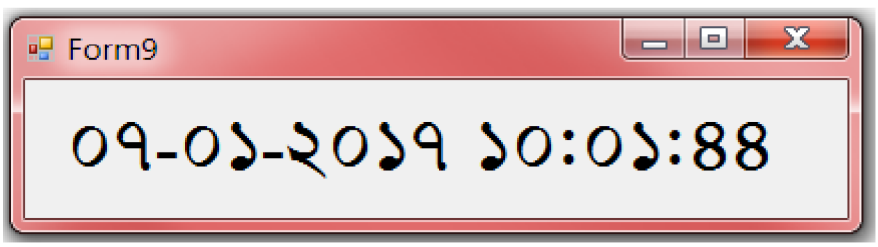
ফর্মের কোথাও ডবল ক্লিক করে এর লোড ইভেন্ড সাবরুটিন প্রদর্শন করুন। এতে লিখুন নিচের কোড লাইনটি। আমরা চাই প্রোগ্রাম শুরু হলেই টাইমার সক্রিয় হয়ে ওঠবে- তাই এই কোড।

```
Private Sub Form9_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Timer1.Enabled = True
End Sub
```

এবার টাইমারে ডবল ক্লিক করে তার ইভেন্ট সাব বের করুন। লিখুন এই কোডটুকু।

```
Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
   Label1.Text = Date.Now.ToString("dd-MM-yyyy hh:mm:ss")
   'Other date format "dd MM yyy hh mm ss"
End Sub
```

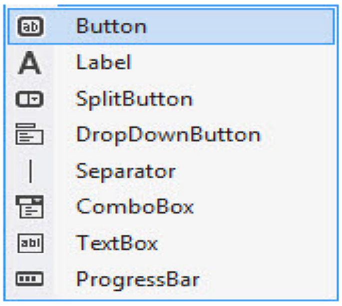
এ কোডের ওপর কিছুটা ব্যাখ্যার প্রয়োজন। আমরা লেবেলে তারিখ ও সময় দেখাতে চাই। সুতরাং এর টেক্সট-এ এখনকার তারিখ ও সময় [Now] প্রদর্শন করতে নির্দেশ দিচ্ছি। তারিখ ও সময় মূলত অবজেক্ট। ওগুলোকে স্ট্রিং বানাতে যেয়ে লিখতে হয়েছে ToSting কীওয়ার্ডটি। ব্রাকেটে আমরা তারিখ ও সময় ফরমাট কি হবে সেটা উল্লেখ করেছি। এবার রান করুন। নিচের চিত্রে আমরা ডিজিটাল ঘড়িটি দেখতে পাচ্ছি- আর এটি বাংলা সংখ্যা প্রদর্শন করছে!



১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

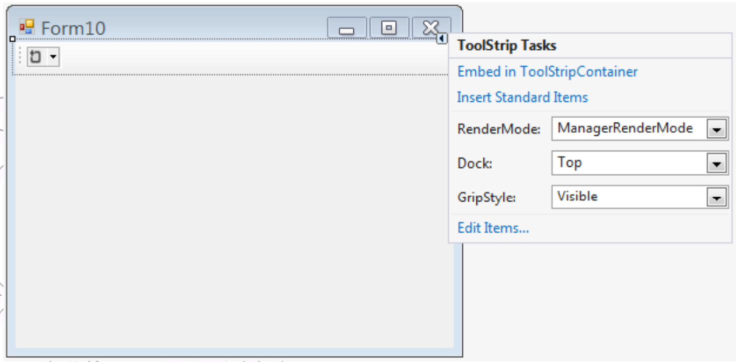
59. ToolStrip: এটার কাজ হলো একাধিক বোতাম, ড্রপডাউন এবং অন্যান্য টুল্স প্রদর্শন করা। এসব টুল্স সহজে ও তাড়াতাড়ি পেতে টুলস্ট্রিপ খুব উপযোগী। বার বার মেনু আইটেমে যাওয়ার প্রয়োজন নেই। নিচের লিস্টে দেখানো হয়েছে টুলস্ট্রিপে কোন্ কোন্ আইটেম থাকতে পারে।

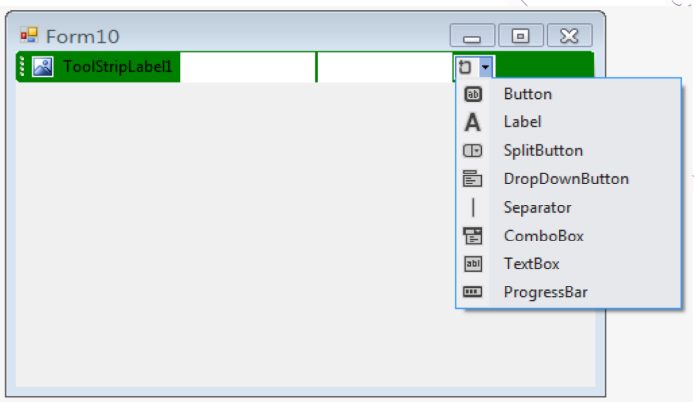


টুলস্ট্রিপ সৃষ্টির দু'টি সময় আছে: ১. ডিজাইনটাইম ও ২. রানটাইম। প্রথমে আমরা ডিজাইটাইমে টুলস্ট্রিপ তৈরির পদ্ধতি দেখবো।

একটি নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে টুলবাক্স থেকে ডবল ক্লিক করে একটি টুলস্ট্রিপ যুক্ত করুন। এটা ফর্মের উপরে যেয়ে ডক করবে। ডানের ছোট্ট এ্যারোতে ক্লিক করে দেখতে পাবেন আরেকটি জানালা। নিচের চিত্রটি দেখুন।

এবার টুলস্ট্রিপে ক্লিক করে এর প্রোপার্টিতে যান। সেট BackColor - সবুজ এবং GripStyle - ভিজিবল। এরপর টুলস্ট্রিপের বায়ের ছোট্ট এ্যারোতে ক্লিক করুন। একটি লিস্ট আসবে [উপরের ছবি]। এ লিস্ট থেকে যে কোনো আইমে চয়েজ করতে পারেন। দেখুন নিচের চিত্রটি।

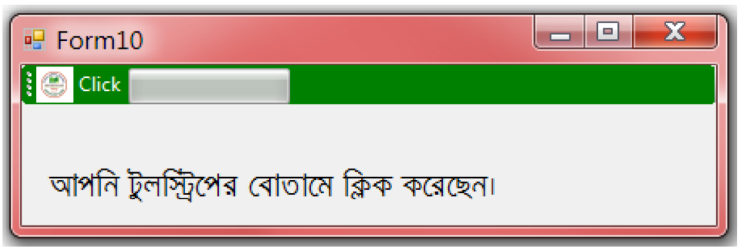




টুলস্ট্রিপ মূলত বিভিন্ন টুলস্ রাখার স্থান। আপনি এগুলোর ক্লিক ও অন্যান্য ইভেন্ট খুলে কোডিং করতে পারেন। যেমন নিচের কোডগুলো হলো যথাক্রমে বোতাম, লেবেল ও প্রগ্রেস বারের ক্লিক ইভেন্ট কোড। প্রজেক্ট রান করে টুলস্ট্রিপের যে কোন আইটেমে ক্লিক করুন।

```
Private Sub ToolStripButton1_Click(sender As Object, e As EventArgs) Handles ToolStripButton1.Click
    Label1.Text = "আপনি টুলস্ট্রিপের বোতামে ক্লিক করেছেন।"
End Sub
Private Sub ToolStripLabel1_Click_1(sender As Object, e As EventArgs) Handles ToolStripLabel1.Click
    Label1.Text = "আপনি টুলস্ট্রিপের লেবেলে ক্লিক করেছেন।"
End Sub
Private Sub ToolStripProgressBar1_Click(sender As Object, e As EventArgs) Handles ToolStripProgressBar1.Click
    Label1.Text = "আপনি টুলস্ট্রিপের প্রগ্রেস বারে ক্লিক করেছেন।"
End Sub
```

মোটকথা, আমরা চাইলে টুলস্ট্রিপে বিভিন্ন আইটেম রেখে এগুলোর বিভিন্ন ইভেন্টে কোড লিখে কাঙ্ক্ষিত ফলাফল পেতে পারি। এভাবে করলে ব্যবহারকারীর জন্য সুন্দর একটি ইন্টারফেইসও তৈরি হতে পারে।

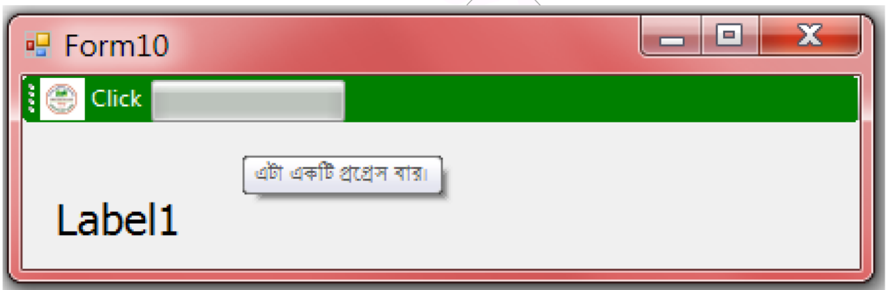


১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

60. ToolStripContainer: এটা একটি কন্টেইনার যার ভেতর ইতোমধ্যে আলোচিত টুলস্ট্রিপ থাকতে পারে। সুতরাং এটা ডিজাইনকালীন সুবিধার জন্য তৈরি হয়েছে।

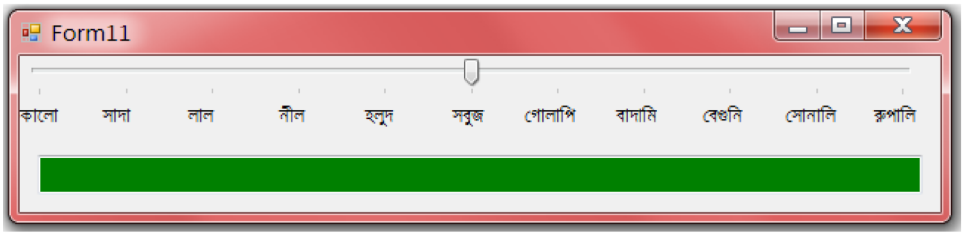
61. ToolTip: এ বস্তুটি খুব ব্যবারোপযোগী। অন্য বস্তু যেমন বোতাম, লেবেল ইত্যাদির উপর মাউস পয়েন্টার গেলেই কিছু লেখা আসবে, যাতে নির্দেশ থাকবে এ বস্তুটির উদ্দেশ্য কি। শিক্ষার্থীরা নিশ্চয়ই টুলটিপ দেখেছেন। দেখে নিই একটি উদাহরণ। আগের টুলস্ট্রিপ সম্পর্কিত প্রোগ্রামে আমরা টুলটিপ কন্ট্রোল যোগ করতে পারি। এবার দেখবেন ফর্মের সকল অন্যান্য কন্ট্রোলের প্রোপার্টিতে 'ToolTipText' নামক একটি অতিরিক্ত আইটেম এসেছে। আপনি এখানে যাই লিখবেন তা-ই আসবে যখনই মাউসটি ঐ কন্ট্রোলের ওপর যাবে। দেখুন নিচের চিত্রটি।



62. TrackBar: এটা অনেকটাই স্ক্রলবারের মতো। তবে এ ক্ষেত্রে ব্যবহারকারী ইচ্ছে মতো যে কোনো স্থানে স্ক্রল করতে পারেন ও সে মুতাবিক কোড করে বিভিন্ন ক্রিয়া সম্পাদনও সম্ভব। একটি সুন্দর উদাহরণ দেখে নিই। প্রথমে আমরা একটি ফর্ম ডিজাইন করবো। দেখুন নিচের রানকালীন চিত্রটি।

কোডিং: এটা খুব সহজ। প্রথমে ট্রাকবার১ এর ওপর ডবল ক্লিক করে কোড জানালা খুলুন। এরপর নিচের কোডগুলো লিখুন। আপনি সব টাইপ করতে হবে না। একটি করে বাকিগুলো কপি-পেস্ট করতে পারেন। পরিবর্তন হবে সংখ্যা ও রঙ।

```
Private Sub TrackBar1_Scroll(sender As Object, e As EventArgs) Handles TrackBar1.Scroll
    Dim tValue As Integer = TrackBar1.Value
    If tValue = 0 Then TextBox1.BackColor = Color.Black
    If tValue = 1 Then TextBox1.BackColor = Color.White
    If tValue = 2 Then TextBox1.BackColor = Color.Red
    If tValue = 3 Then TextBox1.BackColor = Color.Blue
    If tValue = 4 Then TextBox1.BackColor = Color.Yellow
    If tValue = 5 Then TextBox1.BackColor = Color.Green
    If tValue = 6 Then TextBox1.BackColor = Color.Pink
    If tValue = 7 Then TextBox1.BackColor = Color.Brown
    If tValue = 8 Then TextBox1.BackColor = Color.Purple
    If tValue = 9 Then TextBox1.BackColor = Color.Gold
    If tValue = 10 Then TextBox1.BackColor = Color.Silver
End Sub
```



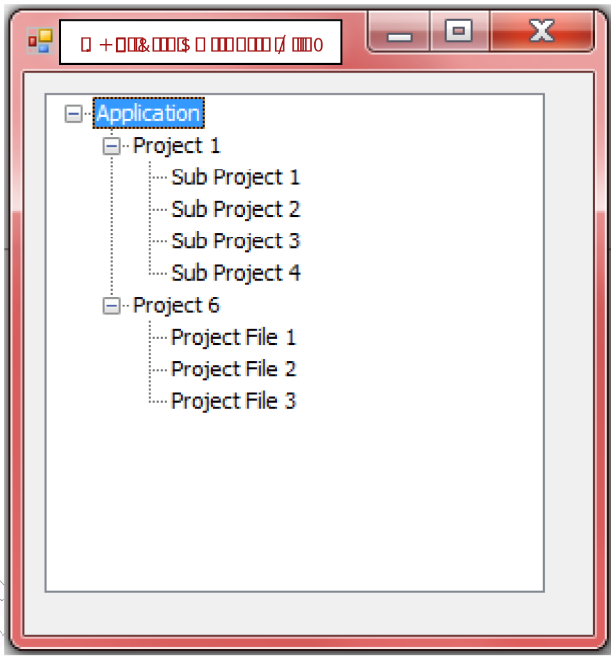
63. TreeView: এ কন্ট্রোল দ্বারা উইন্ডোজের এক্সপ্লোরারের মতো বিভিন্ন বস্তু দেখানো যায়। প্রতিটি নোডে থাকতে পারে আরো বাচ্চা নোড। একটি সাধারণ উদাহরণ থেকে এ সম্পর্কে আরো জানা যাবে। ট্রিভিউ কন্ট্রোল ডিজাইনটাইমে ফর্মে নেওয়া যায় বা প্রোগ্রাম ডোক দ্বারাও তৈরি করা যায়। আমরা দ্বিতীয় পদ্ধতিতে এটি তৈরি করে দেখাবো।

প্রথমেই একটি নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে ডবল ক্লিক করে এর কোড এডিটর জানালা খুলুন। এরপর পরের পৃষ্ঠার কোডগুলো ফর্মের লোড ইভেন্ট সাবে লিখুন।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    'create a new TreeView
    Dim TreeView1 As TreeView
    TreeView1 = New TreeView()
    TreeView1.Location = New Point(10, 10)
    TreeView1.Size = New Size(250, 250)
    Me.Controls.Add(TreeView1)
    TreeView1.Nodes.Clear()
    'Creating the root node
    Dim root = New TreeNode("Application")
    TreeView1.Nodes.Add(root)
    TreeView1.Nodes(0).Nodes.Add(New TreeNode("Project 1"))
    'Creating child nodes under the first child
    For loopindex As Integer = 1 To 4
      TreeView1.Nodes(0).Nodes(0).Nodes.Add(New _
        TreeNode("Sub Project" & Str(loopindex)))
    Next loopindex
    ' creating child nodes under the root
    TreeView1.Nodes(0).Nodes.Add(New TreeNode("Project 6"))
    'creating child nodes under the created child node
    For loopindex As Integer = 1 To 3
      TreeView1.Nodes(0).Nodes(1).Nodes.Add(New _
        TreeNode("Project File" & Str(loopindex)))
    Next loopindex
    ' Set the caption bar text of the form.
    Me.Text = "ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং"
  End Sub
```

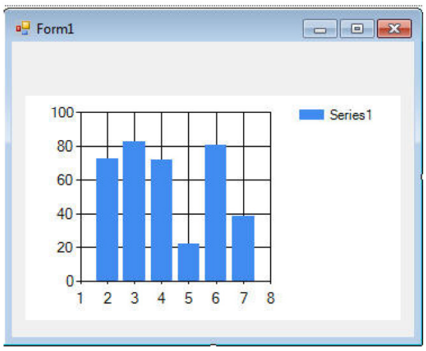


ডানের চিত্রে আমরা ট্রিভিউটি দেখতে পাচ্ছি।

64. VScrollBar: এটা হলো উপর-নিচ স্ক্রলবার কন্ট্রোল। এই কন্ট্রোল সম্পর্কে আমরা ইতোমধ্যে আলোচনা করেছি। দেখুন, পৃষ্ঠা ১৭৮।

65. WebBrowser: এই কন্ট্রোলটি হলো ওয়েভ ব্রাউজার তৈরির ক্লাস। আমরা একটি পূর্ণাঙ্গ ওয়েভ ব্রাউজার তৈরি করেছি। সুতরাং এ সম্পর্কে অতিরিক্ত কিছু বলার নেই। আপনি আবার দেখতে পারেন পৃষ্ঠা ১৬৩ থেকে ১৬৭।

66. Chart [Inside Data Sub Folder]: মনে করুন মাইক্রসফট এক্সেসের একটি ডাটাবেইজ আপনি তৈরি করলেন। সেখানে আছে দু'টি টেবিল: ১. Sales এবং ২. Expenses। আপনি চান, এই টেবিলদ্বয়ের একটি গ্রাফিক্যাল চার্ট আপনার বসকে দেখাবেন। এ কাজের জন্য ব্যবহার করতে পারেন ভিজুয়্যাল বেসিক.নেট এর Chart নামক কন্ট্রোলটি। আমরা এই উদাহরণ প্রোগ্রামটি করে নেবো।

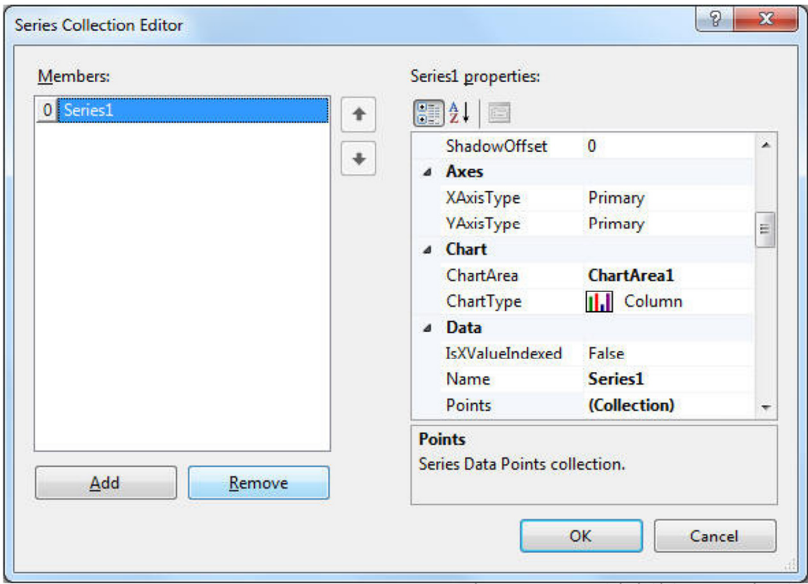


প্রথমে একটি প্রজেক্ট তৈরি করুন। এতে একটি মাত্র বোতাম টুলবাক্স থেকে নিয়ে আসুন। এর নামকরণ করতে পারেন: চার্ট তৈরি করুন। এবার টুলস্‌ বাক্স থেকে একটি চার্ট নিয়ে আসুন ফর্মে। টুলস্‌ বাক্সের Data নামক টাবটিতে ক্লিক করলেই Chart আইটেমটি দেখতে পাবেন। সাধারণত চার্টটি ডানের চিত্রের মতো প্রদর্শিত হবে ফর্মে। তবে আমরা এভাবে রাখবো না। আমরা সেল্‌স ও এক্সপেন্স কথাগুলো লিখবো। সুতরাং Series1 মুছে ফেলতে হবে।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

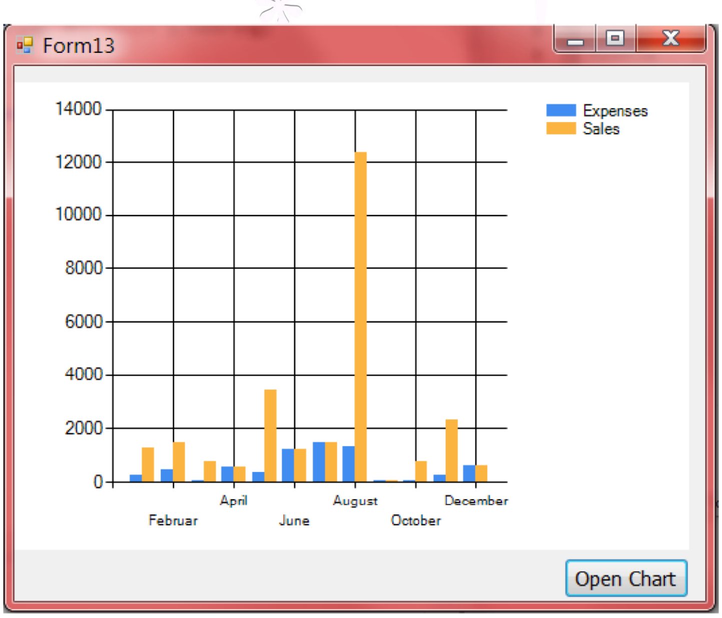
উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

প্রথমে চার্টে যেয়ে রাইট-ক্লিক করুন। চয়েজ করুন প্রোপার্টি। Series -এ যেয়ে ডানের [...] বোতামে ক্লিক করুন। যে জানালা আসবে তা নিচে দেখানো হয়েছে। Series সিলেক্ট করে Remove বোতামে ক্লিক করুন। ব্যস! অকে ক্লিক করে জানালা বন্ধ করুন। এবার 'চার্ট তৈরি করুন' বোতামে ডবল ক্লিক করে কোড এডিটর খুলুন। বোতামের ক্লিক ইভেন্টে নিচের কোডগুলো লিখুন।




```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Chart1.Series.Add("Expenses")
    Chart1.Series.Add("Sales")
    Dim Conn As OleDbConnection = New OleDbConnection
    Dim provider = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source ="
    Dim dataFile = "C:\Users\htci\Desktop\Sales.accdb" ' Change it to your Access Database location
    Conn.ConnectionString = provider & dataFile
    Conn.Open()
  Dim cmd As OleDbCommand = New OleDbCommand("SELECT [Month], [Amount] FROM [Expenses]", Conn)
  Dim dr As OleDbDataReader = cmd.ExecuteReader
  While dr.Read
    Chart1.Series("Expenses").Points.AddXY(dr("Month").ToString, dr("Amount").ToString)
  End While
  dr.Close()
  cmd.Dispose()
  cmd = New OleDbCommand("SELECT [Month], [Amount] FROM [Sales]", Conn)
  dr = cmd.ExecuteReader
  While dr.Read
    Chart1.Series("Sales").Points.AddXY(dr("Month").ToString, dr("Amount").ToString)
  End While
End Sub
```

প্রথমেই আমরা চার্টের সিরিজের মধ্যে দু'টি কথা লিখেছি। এগুলো চার্টের ডানে লিখা থাকবে। এর পরের লাইনে ডাটাবেইজের সঙ্গে যোগাযোগ সৃষ্টি করার জন্য নতুন কানেকশন ডিক্লায়ার করেছি। এরপর এক্সেসের ডাটাবেইজ খোলার যাবতীয় কোড এসেছে। যেখানে লেখা আছে: ' Change it to your Access Database location সেখানে আপনার এক্সেস ফাইলের প্যাথ স্ট্রিং লিখবেন। ব্যস! এই কোড রান করলেই আপনার ডাটাবেইজের চার্ট তৈরি হবে। দেখুন ডানের চিত্রটি।



১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

67. User Control: এতোক্ষণ আমরা ভিজুয়্যাল বেসিক.নেট এর তৈরি করা কন্ট্রোলগুলোর ওপর বিস্তারিত আলেচনা করেছি। তবে এখানেই শেষ নয়। ব্যাবহারকারী বা প্রোগ্রামাররা নিজেরাই নতুন কন্ট্রোল তৈরি করতে পারেন। এগুলোর মাধ্যমে হয়তো বিশেষ কোন কাজও হতে পারে। সুতরাং, আমরা এখন একটি ইউজার কন্ট্রোল তৈরির প্রক্রিয়ার ওপর আলেচনা করবো। বাস্তবে একটি নতুন কন্ট্রোল বানাবো।

মূলত তিন ধরনের ইউজার কন্ট্রোল আছে। যথা:

১. **Combination control:** এর অর্থ একাধিক ফর্ম কন্ট্রোল দ্বারা একটি নতুন কন্ট্রোল তৈরি করা। সকল উইন্ডোজ কন্ট্রোল এটির ভেতরে থাকবে এবং সবগুলো ফাংশন্যাল হবে।

২. **Inherited control:** এটা হলো এমন একটি কন্ট্রোল যা কোনো উইন্ডোজ কন্ট্রেল থেকে উত্তরাধিকার হিসাবে প্রাপ্ত। তবে এতে আরো থাকবে নতুন কিছু ফাংশন।

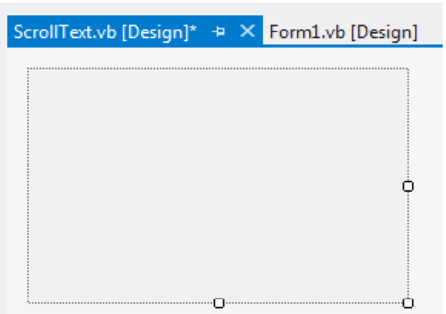
৩. **Custom control:** এটা সম্পূর্ণ নতুন কন্ট্রোল। এতে কোনো রেডিমেইড কন্ট্রোল থাকবে ন।

উপরে বর্ণিত তিনটির মধ্যে প্রথমটির ওপর একটি উদাহরণ এবার তুলে ধরছি। শিক্ষার্থীরা একে একে নিচের পদক্ষেপগুলো অনুসরণ করুন।

১. একটি ফর্ম প্রজেক্ট তৈরি করুন। এর নামকল যাচ্ছেতাই দিতে পারেন।

২. লোড হওয়ার পর চয়েজ করুন: Project > Add User Control। এর নামকরণ করুন: **ScrollText** ।

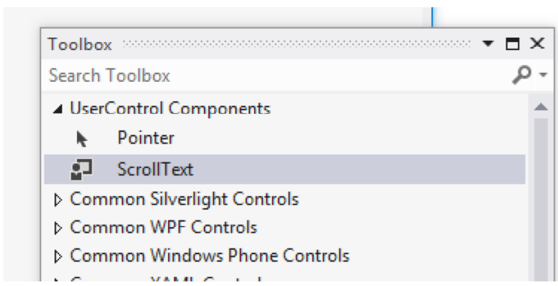
৩. এটি আসবে নিচের চিত্রের মতো। টুলবাক্স থেকে একটি লেবেল কন্ট্রোলে নিয়ে আসুন। এর নামকরণ করুন: LabelScroll। এরপর যুক্ত করুন একটি টাইমার। টাইমারের ওপর ডবল ক্লিক করে এর Tick ইভেন্ট সাব খুলুন। এই সাবের ভেতর লিখুন নিচের কোডগুলো।




```
Public Class ScrollText
    Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
        Dim strBegin As String = Mid(LabelScroll.Text, 1, 1) 'Extract First Character
        Dim strEnd As String = Mid(LabelScroll.Text, 2, LabelScroll.Text.Length()) 'Extract Rest Of String
        LabelScroll.Text = strEnd & strBegin 'Add Beginning To End
    End Sub
End Class
```

লক্ষ করুন, এমনিতেই আমাদের কন্ট্রোলের নামে একটি ক্লাস এসেছে, ঠিক যেরূপ কোনো ফর্মের ক্লাস সাবরুটিন আসে। এই কোড একটি চলন্ত স্ট্রিং [লেখা] তৈরি করে। লেখার প্রথম অক্ষর ও বাকীগুলো যোগ করে এই কাজটি আঞ্জাম দেয়।

৪. এবার প্রোগ্রামটি বিল্ড করুন: Menu > Build > Build Solution। বিল্ড শেষে লক্ষ করুন, টুলবাক্সে আমাদের সৃষ্ট ScrollText কন্ট্রোলটিও User Control Components লিস্টে এসে গেছে। দেখুন নিচের চিত্রটি।



৫. এবার এই নতুন কন্ট্রোলটি ডবল ক্লিক করে আমাদের উইন্ডোজ ফর্মের ভেতর নিয়ে আসুন।

৬. ফর্মে দু'টি বোতাম যুক্ত করুন। এদের টেক্সট হবে যথাক্রমে: "শুরু" ও "বন্ধ"।

৭. 'শুরু' বোতামের ক্লিক সাবরুটিনে এই দু'টি লাইন লিখুন:

```
ScrollText1.LabelScroll.Text = "আমি তো ভাই খুব বেশি বেশি বেশি ক্ষুধার্ত..." 'Provide Text
ScrollText1.Timer1.Enabled = True 'Enable Timer
```

৭. 'বন্ধ' বোতামের ক্লিক সাবরুটিনে এই একটি লাইন লিখুন:

```
ScrollText1.Timer1.Enabled = False 'Stop Scrolling
```

ব্যস! আমাদের সৃষ্ট কন্ট্রোল এখন কাজ করবে। রান করে দেখুন।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

এটাই গ্রন্থের সর্বশেষ পরিচ্ছেদ। ভিজুয়্যাল বেসিক.নেট সম্পর্কে আমার যাকিছু শিখেছি তার ওপর সহজ একটি গাইড এ পরিচ্ছেদে তুলে ধরবো। শিক্ষার্থীরা নিজেদের এ্যাপলিকেশন সৃষ্টিতে এই পরিচেচ্ছদকে সময় সময় কাজে লাগাতে পারবেন। এছাড়া মাইক্রসফটের এই শকিক্তশালী প্রোগগ্রামিং ভাষার ওপরও এ গাইড থেকে অধিক বেশি জ্ঞানবান হবেন- এটাই আমার বিশ্বাস।

১. সার্বিক ধারণা: Overview

Microsoft Visual Basic.Net একটি অবজেক্ট অরিয়েন্টেড প্রোগগ্রামিং ভাষা। এর বাস্তবায়ন .নেট [বর্তমান ভার্শন: ৪.৫] নামক ফ্রেমওয়ার্কের ওপর নির্ভরশীল। ভাষার সবকিছুই মূলত বস্তুভিত্তিক। এর মৌলিক টাইপও [Short, Long, Integer, String, Boolean etc.] একেকটি বস্তু। .নেট দ্বারা সৃষ্ট প্রোগ্রাম উইনেন্ডাজ ছাড়াও লাইনাক্স [Linux] কিংবা ম্যাক অএসএক্স [Mac OSX] অপরেটিং সিস্টেমে রান করা যায়।

২. ডটনেট ফ্রেমওয়ার্ক: .Net Framework

এটা মাইক্রসফটের একটি বৈপ্লবিক সর্বাধুনিক প্রোগ্রামিং ফ্ল্যাটফর্ম। এটার দ্বারা প্রোগ্রামাররা সমগ্র বিশ্বব্যাপী প্রতিনিয়ত এ্যাপলিকেশন সৃষ্টি করে যাচ্ছেন:
(ক) উইন্ডোজ ডেস্কটপ এ্যাপলিকেশন।
(খ) ওয়েভ এ্যাপলিকেশন ও সার্ভিসেস।

ডটনেট এ্যাপলিকেশন বহুমুখি-ফ্লাটফর্মও বটে। এটি ডিজাইন করা হয়েছে বেশ ক'টি সফটওয়্যার ভাষার কথা চিন্তা করে। এগুলোর মধ্যে Visual Basic, C#, C++, Jscript, and COBOL ইত্যাদি এখানে উল্লেখযোগ্য। এ ভাষাগুলো ডটনেট ফ্রেমওয়ার্ক কাজে লাগাতে পারে এবং একে অন্যের সঙ্গে যোগাযোগও সৃষ্টি করতে পারে। ফ্রেমওয়ার্কের মধ্যে একটি বিরাট কোড লাইব্রেরি আছে যা উক্ত ভাষাগুলোর মাধ্যমে ব্যবহার করা যায়। এই গ্রন্থে আলোচিত Vb.Net মূলত লাইব্রেরির কোড ব্যবহারেরই প্রশিক্ষণ মাত্র।

৩. প্রোগ্রাম কাঠামো : Program Structure

যে কোন প্রোগ্রামে সাধারণত এই বিষয়গুলো থাকে: ক. নেইমস্পেইস ঘোষণা [Namespace declaration] উদাহরণ: Imports System, খ. ক্লাস বা মডিউল [Class or module] উদাহরণ: Public Class Form1; Module Module1, গ. প্রসিডিউর [Procedures- Sub, Function, Operator, Get, Set, AddHandler, RemoveHandler, RaiseEevent], ঘ. ভেরিয়েবল [Variables] উদাহরণ: Dim UserName as String, ঙ. মূল প্রসিডিউর [Main procedure] উদাহরণ: Sub Main(), চ. এক্সপ্রেশন ও স্টেটমেন্ট [Expressions & Statements] উদাহরণ: If MyVar = "Book" then; For x = 1 to 100 এবং ছ. মন্তব্য [Comments] উদাহরণ: 'This is the global variable for keeping User Name। শেষোক্ত [ছ] বিষয়টি বাস্তবে প্রোগ্রামের সঙ্গে সম্পৃক্ত নয়। এটা কম্পাইলার উপেক্ষা করে যাবে। তবে কমেন্ট থাকলে প্রোগ্রাম বুঝা অনেকটা সহজ হয়। ডিবাগিং ও নতুন সংস্করণ প্রকাশে এগুলো খুব সহায়ক হয়ে থাকে।

৪. ভিজুয়্যাল বেসিক.নেট মৌলিক সিনটেক্সট: VB.Net - Basic Syntax

আগেই বলেছি ভিজুয়্যাল বেসিক.নেট একটি পূর্ণাঙ্গ অবজেক্ট ওরিয়েন্ডেড ভাষা। এর অর্থ হলো কোনো প্রোগ্রামে থাকবে বিভিন্ন বস্তু বা অবজেক্ট। তারা একে অন্যের সঙ্গে মিথস্ক্রিয়া করবে কিছু ক্রিয়ার মাধ্যমে। আর এসব ক্রিয়ার নামই হচ্ছে পদ্ধতি বা মেথড। একই ধরনের অবজেক্টকে বলে একই টাইপ বা ক্লাসের। সুতরাং আমরা যখন কোনো ভিজুয়্যাল বেসিক.নেট প্রোগ্রাম নিয়ে আলোচনা করি তখন একে সংজ্ঞায়িত করতে পারি এভাবে: একগুচ্ছ অবজেক্ট বা বস্তু যারা একে অন্যের সঙ্গে যোগাযেগ সৃষ্টি করে যার তার মেথড বা পদ্ধতিসমূহের মাধ্যমে। এসব বস্তু ও মেথডের ওপর আমরা আরো

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

কিছু ব্যাখ্যা তুলে ধরতে পারি।
(ক) অবজেক্ট [Object]: অবজেক্টের আছে হাল বা স্টেট [State] ও চরিত্র [behavior]। উদাহরণ: একটি কুকুরের হালসমূহের মাধ্যে আছে: রং, নাম, ধরন। আরো আছে তার চরিত্র: লেজ নাড়ানো, প্রভুভক্তি, হাড্ডি খাওয়া, ডাকা, ছুটোছুটি করা, পান করা ইত্যাদি। অবজেক্ট হলো ক্লাসের ইন্সট্যান্ট [মুহূর্ত]।
(খ) ক্লাস [Class]: ক্লাসকে আমরা সংজ্ঞায়িত করতে পারি এভাবে: একটি ব্লু প্রিন্ট বা টেমপ্লেট যা ব্যাখ্যা করে এই টাইপের বস্তুর অবস্থা ও চরিত্র।
(গ) মেথড [Method]। এটা মূলত একটি চরিত্রকে বুঝায়। ক্লাসের মধ্যে অনেকগুলো মেথড থাকতে পারে। আর এসব মেথডের মধ্যেই আমরা লজিক বা যুক্তিগুলো লিখে থাকি। এগুলোর মধ্যেই তথ্যাদি পরিবর্তন, পরিবর্ধন, কাজে লাগানো ও সকল ক্রিয়া সম্পাদিত হয়।
(ঘ) মুহূর্ত ভেরিয়েবল [Instant variables]: প্রতিটি অবজেক্টেরই আছে তার নিজস্ব এক বা ততোধিক ভেরিয়েবল। এই মুহূর্ত ভেরিয়েবলসমূহে [অর্থাৎ অস্থায়ী কন্টেইনারে] ভ্যালু বা মান নির্দিষ্ট [assignment] করেই অবজেক্টের State সৃষ্টি করা হয়।

৫. পরিচায়ক: Identifiers

প্রোগ্রামার দ্বারা ডিফাইন করা যে কোনো ক্লাস, ভেরিয়েবল, ফাংশন, সাবরুটিন কিংবা যে কোনো বস্তুর পরিচায়ক হিসাবে বিশেষ নাম থাকতে হয়। এ নামকেই বলে আইডেন্টিফাইয়ার। নামগুলো হতে হবে একক। এছাড়া ভিজুয়্যাল বেসিক.নেট ভাষার কিছছু নাম আছে যেগগুলো রিজর্ভ। এগুলো প্রোগ্রামে আইডেন্টিফাইয়ার হিসাবে ব্যবহার করা নিষিদ্ধ। আমরা একটু পরই রিজার্ভ কিওয়ার্ডগগুলোর একটি লিস্ট তুলে ধরছি। আইডেন্টিফাইয়ার কিভাবে লিখবেন তারও কিছু নিয়মকানুন আছে। আমরা সেগুলো প্রথমে উল্লেখ করছি।
ক. নামের প্রথম অক্ষর কোনো সংখ্যা দেওয়া যাবে না। তবে আন্ডারস্কোর [_] গ্রহণযোগ্য। নামের মাঝখানে শূন্যস্থান [স্পেইস] দেওয়া যাবে না। যেমন: Dim _MyVariable; Dim FirstNumber1 - উভয়টি গ্রহণযোগ্য। কিন্তু, Number One কিংবা Dim 3TheNumber- গ্রহণযোগ্য নয়।
খ. আইডেন্টিফাইয়ারে এই ক্যারাক্টারগুলো দেওয়া যাবে না: ? - +! @ # % ^ & * ( ) [ ] { } . ; : " ' / \ | ইত্যাদি।

নিচের টেবিলে আমরা রিজার্ভ কিওয়ার্ডগুলো তুলে ধরেছি।

AddHandler	AddressOf	Alias	And	AndAlso	As	Boolean
ByRef	Byte	ByVal	Call	Case	Catch	CBool
CByte	CChar	CDate	CDec	CDbl	Char	CInt
Class	CLng	CObj	Const	Continue	CSByte	CShort
Decimal	Declare	Default	Delegate	Dim	DirectCast	Do
Double	Each	Else	ElseIf	End	End If	Enum
Erase	Error	Event	Exit	False	Finally	For
Friend	Function	Get	GetType	GetXML Namespace	Global	GoTo
Handles	If	Implements	Imports	In	Inherits	Integer
Interface	Is	IsNot	Let	Lib	Like	Long
Loop	Me	Mod	Module	MustInherit	MustOverride	MyBase
MyClass	Namespace	Narrowing	New	Next	Not	Nothing
Not Inheritable	Not Overridable	Object	Of	On	Operator	Option
Optional	Or	OrElse	Overloads	Overridable	Overrides	ParamArray

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

Partial	Private	Property	Protected	Public	RaiseEvent	ReadOnly
ReDim	REM	Remove Handler	Resume	Return	SByte	Select
Set	Shadows	Shared	Short	Single	Static	Step
Stop	String	Structure	Sub	SyncLock	Then	Throw
To	True	Try	TryCast	TypeOf	UInteger	While
Widening	With	WithEvents	WriteOnly	Xor		

উপরোক্ত টেবিলের সকল শব্দ মূলত ভিজুয়্যাল বেসিক.নেট ভাষার সঙ্গে সম্পৃক্ত। সুতরাং এগুলো আইডেন্টিফাইয়ার হিসাবে ব্যবহার করা যাবে না।

৬. তথ্যাদির ধরন: Data Types

সকল প্রোগ্রামিং ভাষার মতো ভিজুয়্যাল বেসিক.নেটে আমরা সব ধরনের ডাটা টাইপ ব্যবহার করতে পারি। গ্রন্থের শুরুর দিকে আমরা এসব টাইপ নিয়ে আলোচনা করেছি। এখানে শুধুমাত্র লিস্টটি পূর্ণাঙ্গভাবে তুলে ধরার প্রয়োজন মনে করছি।

Data Type	Storage Allocation	Value Range
Boolean	Depends on implementing platform	**True** or **False**
Byte	1 byte	0 through 255 (unsigned)
Char	2 bytes	0 through 65535 (unsigned)
Date	8 bytes	0:00:00 (midnight) on January 1, 0001 through 11:59:59 PM on December 31, 9999
Decimal	16 bytes	0 through +/- 79,228,162,514,264,337,593,543,950,335 (+/- 7.9...E+28) with no decimal point; 0 through +/ -7.9228162514264337593543950335 with 28 places to the right of the decimal
Double	8 bytes	-1.79769313486231570E+308 through -4.94065645841246544E-324, for negative values 4.94065645841246544E-324 through 1.79769313486231570E+308, for positive values
Integer	4 bytes	-2,147,483,648 through 2,147,483,647 (signed)
Long	8 bytes	-9,223,372,036,854,775,808 through 9,223,372,036,854,775,807(signed)
Object	4 bytes on 32-bit platform 8 bytes on 64-bit platform	Any type can be stored in a variable of type Object
SByte	1 byte	-128 through 127 (signed)
Short	2 bytes	-32,768 through 32,767 (signed)
Single	4 bytes	-3.4028235E+38 through -1.401298E-45 for negative values; 1.401298E-45 through 3.4028235E+38 for positive values

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

Data Type	Storage Allocation	Value Range
String	Depends on implementing platform	0 to approximately 2 billion Unicode characters
UInteger	4 bytes	0 through 4,294,967,295 (unsigned)
ULong	8 bytes	0 through 18,446,744,073,709,551,615 (unsigned)
User-Defined	Depends on implementing platform	Each member of the structure has a range determined by its data type and independent of the ranges of the other members
UShort	2 bytes	0 through 65,535 (unsigned)

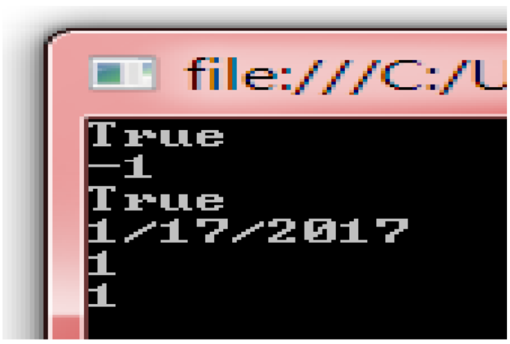
৭. টাইপ রূপান্তর ফাংশন: Types Conversion Functions

ভিজুয়্যাল বেসিক.নেট নিচের টেবিলে উল্লেখিত ফাংশনগুলো প্রোগ্রামারদের সুবিধার্থে তৈরি করে দিয়েছে।

S.N	Functions & Description	S.N	Functions & Description
1	**CBool(expression)** Converts the expression to Boolean data type.	9	**CObj(expression)** Converts the expression to Object type.
2	**CByte(expression)** Converts the expression to Byte data type.	10	**CSByte(expression)** Converts the expression to SByte data type.
3	**CChar(expression)** Converts the expression to Char data type.	11	**CShort(expression)** Converts the expression to Short data type.
4	**CDate(expression)** Converts the expression to Date data type	12	**CSng(expression)** Converts the expression to Single data type.
5	**CDbl(expression)** Converts the expression to Double data type.	13	**CStr(expression)** Converts the expression to String data type.
6	**CDec(expression)** Converts the expression to Decimal data type.	14	**CUInt(expression)** Converts the expression to UInt data type.
7	**CInt(expression)** Converts the expression to Integer data type.	15	**CULng(expression)** Converts the expression to ULng data type.
8	**CLng(expression)** Converts the expression to Long data type.	16	**CUShort(expression)** Converts the expression to UShort data type.

উদাহরণ: নিচের কোড ব্লকটি একটি কনসোল প্রজেক্ট তৈরি করে মডিউলে লিখে রান করুন।

```
Module Module1
    Sub Main()
        Dim n As Integer
        Dim da As Date
        Dim bl As Boolean = True
        n = 1234567
        da = Today
        Console.WriteLine(bl)
        Console.WriteLine(CSByte(bl))
        Console.WriteLine(CStr(bl))
        Console.WriteLine(CStr(da))
        Console.WriteLine(CChar(CChar(CStr(n))))
        Console.WriteLine(CChar(CStr(da)))
        Console.ReadKey()
    End Sub
End Module
```



বায়ের কনসোল প্রোগ্রাম রানের ফলাফল।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

৮. ভিজুয়্যাল বেসিক.নেট ভেরিয়েবল : Visual Basic.Net Variables

ভেরিয়েবল কম্পিউটারের স্টোরেজ মেমোরিতে একটি নাম ছাড়া কিছুই নয়। আমরা এই নামে বিভিন্ন তথ্য সংরক্ষণ করি ও প্রয়োজনে বের করে পরিবর্তন করতে পারি। প্রত্যেক ভেরিয়েবল অবশ্য কোনো বিশেষ ধরনের হতে হবে। একে বলে ভেরিয়েবলের টাইপ। কোন্ টাইপের ভেরিয়েবল তা-ই নির্ধারণ করে কী পরিমাণ মেমোরির প্রয়োজন। ইতোমধ্যে আমার ভেরিয়েবল টাইপ নিয়ে আলোচনা করেছি। মৌলিক টাইপগুলোকে আমরা নিচের টেবিলে দেওয়া পদ্ধতিতে শ্রেণিভুক্ত করতে পারি।

Type	Example
Integral types	SByte, Byte, Short, UShort, Integer, UInteger, Long, ULong and Char
Floating point types	Single and Double
Decimal types	Decimal
Boolean types	True or False values, as assigned
Date types	Date

৯. ভেরিয়েবল ডিক্লারেশন ও ইনিশিয়েলাইজেশন: Variable Declaration and Initialization

শিক্ষার্থীরা ভেরিয়েবল ডিক্লারেশন ও ইনিশিয়েলাইজেশন সম্পর্কে ওয়াকিফহাল আছেন। এরপরও নিচে পুর্ণাঙ্গ নিয়ম-কানুন তুলে ধরছি রেফারেন্স হিসাবে।

Syntax: [< attribute list>] [access modifier] [[Shared] [Shadows] | [Static]] [ReadOnly] Dim [WithEvents] variable list

উপরের সিনট্যাক্সের প্রতিটি শব্দের ব্যাখ্যা:

attribute list: এটা হলো ভেরিয়েবলের সঙ্গে সম্পর্কিত গুণাবলীর তালিকা- এটা অপশন্যাল।

access modifier: ভেরিয়েবলের প্রাপ্তির ব্যাপ্তি কতটুকু হবে। যেমন: Public, Protected, Friend, Protected Friend and Private। এটাও অপশন্যাল।

Shared: এটা ডিক্লায়ার করে যে ভেরিয়েবলটি সকল স্ট্রাকচার ও ক্লাসে ব্যবহারযোগ্য। এটা অপশন্যাল।

Shadows: এটা নির্দেশ করে যে ভেরিয়েবলটি পুনরায় একই নামে বেইজ ক্লাসে ডিক্লায়ার করা আছে। এটা অপশন্যাল।

Static: যে প্রসিডিউরে এটি ডিক্লায়ার হয়েছে সেটি বন্ধ হওয়ার পরও ভেরিয়েবলটি অপরিবর্তিত থাকবে। এটা অপশন্যাল।

ReadOnly: এর অর্থ ভেরিয়েবলটি শুধুমাত্র পড়া যাবে- কিন্তু লেখা যাবে না। এটা অপশন্যাল।

Dim: এর মানে ডাইমেনশন। এটা বাধ্যতামূলক।

WithEvents: ভেরিয়েবলে যে ইন্সস্ট্যান্ট [মুহূর্ত] এসাইন করা হয়েছে তা এর দ্বারা সক্রিয় হতে পারে। অপশন্যাল।

variable list: এতে থাকে ডিক্লায়ার করা ভেরিয়েবল তালিকা। বাধ্যতামূলক।

প্রত্যেক ভেরিয়েবল লিস্টে থাকবে:

variable name[([bounds list])] [As [New] data type] [= initializer]

variable name: এটা হচ্ছে ভেরিয়বলের নাম। বাধ্যতামূলক।

bounds list: এ্যারে ভেরিয়েবলের প্রত্যেক বিস্তৃতির ব্যাপ্তি ডিক্লায়ার করে। এটা অপশন্যাল।

As: কিওয়ার্ড। এটা থাকতে হবে।

New: এটা ক্লাসের নতুন একটি ইন্সস্ট্যান্ট তৈরি করে যখন ডাইমেনশন [Dim] স্টেটম্যান্ট রান করে। ঐচ্ছিক।

data type: অপশন স্ট্রিক্ট [Option Strict] অন থাকলে এটা বাধ্যতামূলক।

initializer: যদি নিউ [New] লেখা না থাকে, এটা ব্যবহার অপশন্যাল।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

কিছু গ্রহণযোগ্য ডিক্লারেশের উদাহরণ:

```
Dim StudentID As Integer
Dim StudentName As String
Dim Salary As Double
Dim count1, count2 As Integer
Dim status As Boolean
Dim exitButton As New System.Windows.Forms.Button
Dim lastTime, nextTime As Date
```

ভেরিয়েবল ইনিশিয়েলাইজেশন হয় নামের পরে সমান চিহ্ন [=] ও মান লিখে। যেমন: variable_name = value;
উদাহরণ:

```
Dim pi As Double
pi = 3.14159
```

প্রয়োজনে ডিক্লারেশনকালেও আমরা ভেরিয়েবল ইনিশিয়েলাইজ করতে পারি। যেমন:

```
Dim StudentID As Integer = 100
Dim StudentName As String = "Asif Talukder"
```

অপরিবর্তনীয় ভেরিয়েবল ডিক্লারেশন: Declaring Constants : ভিজুয়্যাল বেসিক.নেটে যে কোনো স্থানে ভেরিয়েবলের মতো কন্সস্ট্যান্ট ভেরিয়েবল ডিক্লায়ার করা যায়। এগুলো ঠিক ভেরিয়েবলের মতোই, কিন্ত এদের মান পরিবর্তন হয় না। একটি উদাহরণ দেখি।

```
Module constantsNenum
  Sub Main()
    Const PI = 3.14149 'here is the constant declaration
    Dim radius, area As Single
    radius = 7
    area = PI * radius * radius
    Console.WriteLine("Area = " & Str(area))
    Console.ReadKey()
  End Sub
End Module
```

উপরের প্রোগ্রাম রানের ফলাফল: Area = 153.933।

১০. প্রিন্ট ও প্রদর্শন কন্সস্ট্যান্ট: Print and Display Constants

ভিজুয়্যাল বেসিক.নেট আমাদেরকে নিচের কন্সস্ট্যান্টগুলো তৈরি করে দিয়েছে।

Constant	Description
vbCrLf	Carriage return/linefeed character combination.
vbCr	Carriage return character.
vbLf	Linefeed character.
vbNewLine	Newline character.
vbNullChar	Null character.
vbNullString	Not the same as a zero-length string (""); used for calling external procedures.
vbObjectError	Error number. User-defined error numbers should be greater than this value. For example: Err.Raise(Number) = vbObjectError + 1000
vbTab	Tab character.
vbBack	Backspace character.

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

১১. লিস্ট ডিক্লারেশন: Declaring Enumerations

একসাথে একাধিক ভেরিয়েবল ডিক্লারেশন করার একটি পদ্ধতির নাম ইনুমারেশন **[Enumeration]**। আমরা চাইলে Enum .... End Enum ব্লকের ভেতর ভেরিয়েবল লিস্ট ডিক্লায়ার করতে পারি। যেমন:

```
Module constantsNenum
  Enum Colors
    red = 1
    orange = 2
    yellow = 3
    green = 4
    azure = 5
    blue = 6
    violet = 7
  End Enum
  Sub Main()
    Console.WriteLine("The Color Red is : " & Colors.red)
    Console.WriteLine("The Color Yellow is : " & Colors.yellow)
    Console.WriteLine("The Color Blue is : " & Colors.blue)
    Console.WriteLine("The Color Green is : " & Colors.green)
    Console.ReadKey()
  End Sub
End Module
```

বায়ের কনসোল প্রোগ্রামটি রান করলে আমরা যে ফলাফল পাবো তাহলো:

```
The Color Red is: 1
The Color Yellow is: 3
The Color Blue is: 6
The Color Green is: 4
```

১২. স্টেটমেন্ট: Statements

ইতোমধ্যে এ শব্দটি একাধিকবার উল্লেখিত হয়েছে। ভিজিুয়্যাল বেসিক.নেটে শব্দটির গুরুত্ব বিরাট। কোডের মধ্যে পূর্ণাঙ্গ নির্দেশনার নাম স্টেটমেন্ট। এতে থাকতে পারে: keywords, operators, variables, literal values, constants and expressions । মূলত দু'ধরনের স্টেটমেন্ট আছে: (১) ডিক্লারেশন স্টেটমেন্ট: এগুলো হলো যাদের মধ্যে থাকবে ভেরিয়েবলের নাম, কন্সট্যান্ট বা প্রসিডোর- এতে ডাটা টাইপ ও উল্লেখিত হতে পারে। (২) এক্সেকিউটেবল স্টেটমেন্ট: এগুলো ঐরূপ যারা কোনো ক্রিয়া শুরু করে। এগুলো কোনো মেথড বা ফাংশন কল করতে পারে, কোড ব্লকে লুপিং ও ব্রাঞ্চিং করতে পারে। এ ধরনের স্টেটমেন্ট দ্বারা মান কিংবা এক্সপ্রেশন ভেরিয়েবল বা কন্সট্যান্টে এসাইন করা যায়। আর এরূপ করাকে বলে এসাইনমেন্ট স্টেটমেন্ট। আমরা এই উভয় প্রকার স্টেটমেন্ট এর ওপর এখন আলাদাভাবে ব্যাখ্যা তুলে ধরছি।

(১) ডিক্লারেশন স্টেটমেন্ট: Declaration Statements

এই স্টেটমেন্ট ব্যবহৃত হয় প্রসিডোর, ভেরিয়েবল, প্রোপার্টি, এ্যারে ও কন্সট্যান্ট এর নাম ও ডিক্লারেশনের জন্য। নিচের টেবিলে ভিজুয়্যাল বেসিক.নেট এর ডিক্লারেশন স্টেটমেন্টগুলো তুলে ধরেছি উদাহরণসহ।

S.N	Statements and Description	Example
1	**Dim Statement** Declares and allocates storage space for one or more variables.	Dim number As Integer Dim quantity As Integer = 100 Dim message As String = "Hello!"
2	**Const Statement** Declares and defines one or more constants.	Const maximum As Long = 1000 Const naturalLogBase As Object = CDec(2.7182818284)

পরের পৃষ্ঠায় বাকি টেবিল ... ...

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

S.N	Statements and Description	Example
3	**Enum Statement** Declares an enumeration and defines the values of its members.	Enum CoffeeMugSize Jumbo ExtraLarge Large Medium Small End Enum
4	**Class Statement** Declares the name of a class and introduces the definition of the variables, properties, events, and procedures that the class comprises.	Class Box Public length As Double Public breadth As Double Public height As Double End Class
5	**Structure Statement** Declares the name of a structure and introduces the definition of the variables, properties, events, and procedures that the structure comprises.	Structure Box Public length As Double Public breadth As Double Public height As Double End Structure
6	**Module Statement** Declares the name of a module and introduces the definition of the variables, properties, events, and procedures that the module comprises.	Public Module myModule Sub Main() Dim user As String = InputBox("What is your name?") MsgBox("User name is" & user) End Sub End Module
7	**Interface Statement** Declares the name of an interface and introduces the definitions of the members that the interface comprises.	Public Interface MyInterface Sub doSomething() End Interface
8	**Function Statement** Declares the name, parameters, and code that define a Function procedure.	Function myFunction (ByVal n As Integer) As Double Return 5.87 * n End Function
9	**Sub Statement** Declares the name, parameters, and code that define a Sub procedure.	Sub mySub(ByVal s As String) Return End Sub
10	**Declare Statement** Declares a reference to a procedure implemented in an external file.	Declare Function getUserName Lib "advapi32.dll" Alias "GetUserNameA" (ByVal lpBuffer As String, ByRef nSize As Integer) As Integer
11	**Operator Statement** Declares the operator symbol, operands, and code that define an operator procedure on a class or structure.	Public Shared Operator + (ByVal x As obj, ByVal y As obj) As obj Dim r As New obj ' implemention code for r = x + y Return r End Operator

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

S.N	Statements and Description	Example
12	**Property Statement** Declares the name of a property, and the property procedures used to store and retrieve the value of the property.	ReadOnly Property quote() As String Get Return quoteString End Get End Property
13	**Event Statement** Declares a user-defined event.	Public Event Finished()
14	**Delegate Statement** Used to declare a delegate.	Delegate Function MathOperator(ByVal x As Double, ByVal y As Double) As Double

(২) এক্সেকিউটেবল স্টেটমেন্ট: Executable Statements

এরূপ স্টেটমেন্টের কাজ হলো ক্রিয়া সম্পাদন করা। যেসব স্টেটমেন্ট অন্য প্রসিডোর বা ফাংশন কল করে, কোডের অন্যত্র একশন প্রতিস্থাপন করে, একাধিক স্টেটমেন্টে লুপিং করে ইত্যাদি হচ্ছে এক্সেকিউটেবল স্টেটমেন্ট। আর একটি এসাইনমেন্ট স্টেটমেন্ট হলো এক্সেকিউটেবল স্টেটমেন্টের একটি বিশেষ রূপ। আমরা একটি উদাহরণ দেখতে পারি:

```
Module decisions
  Sub Main()
    'local variable definition '
    Dim a As Integer = 10

    ' check the boolean condition using if statement '
    If (a < 20) Then
      ' if condition is true then print the following '
      Console.WriteLine("a is less than 20")
    End If
    Console.WriteLine("value of a is : {0}", a)
    Console.ReadLine()
  End Sub
End Module
```

বায়ের কোড রান করে যে ফলাফল আসবে, তাহলো:

a is less than 20;
value of a is : 10

১৩. অপারেটর: Operators

গাণিতিক বা যৌক্তিক [লজিক্যাল] কোনো ক্রিয়া সম্পাদনের নির্দেশ দিতে ব্যবহৃত হয় কিছু প্রতীক। এগুলোকেই কম্পিউটার বিজ্ঞানে অপারেটর বলে। ভিজুয়্যাল বেসিক.নেটে প্রচুর ডিফাইন-করা অপারেটর আছে। নিচে তালিকাভুক্ত অপারেটরগুলো সর্বাধিক বেশি ব্যবহৃত হয়।

* পাটিগণিতের প্রতীক [Arithmetic Operators] যেমন: +, -, *, /, Mod, ^, \ [যোগ, বিয়োগ পূরণ ও ভাগ। Mod সর্বদাই ভাগশেষ দেবে। যেমন: ৭ Mod ২= ১। এটা [^] মানে পাওয়ার। সুতরাং ৭ ^ ২ = ৪৯। শেষোক্তটি শুধুমাত্র ইন্টেজার ফলাফল দেবে। যেমন: ৭ \ ২ = ৩]

* পার্থক্য নির্ণয়ক প্রতীক [Comparison Operators] যেমন: = [উভয় দিকের মান সমান], <> [উভয় দিকের মান সমান নয়], > [ডানদিকের মান বামদিক থেকে ছোট], < [ডানদিকের মান বাম দিক থেকে বড়ো], >= [বামদিকের মান ডানদিক থেকে বড়ো না হয় সমান], <= [বামদিকের মান ডানদিক থেকে ছোট বা সমান]

* লজিক্যাল / বিটওয়েজ প্রতীক [Logical/Bitwise Operators] যেমন: And, And এর উভয় দিকের মান সমান থাকলে ফলাফল True। Or, উভয় দিকের যে কোনো একটি সত্য হলে ফলাফল সত্য বা True। Not, কন্ডিশন সত্য হলে এটা একে মিথ্যায় পরিণত করে। উদাহরণ If Not A=10 then- অর্থাৎ A যদি ১০ না হয় তাহলে [সত্য]। ১০ হলে কিন্তু

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

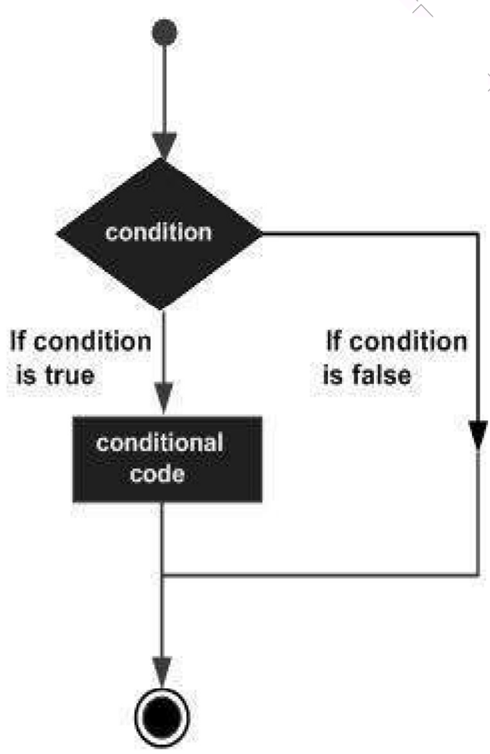
ফলাফল মিথ্যা হবে। Xor -উভয় এক্সপ্রেশন সত্য হলে ফলাফল সত্য [True]। অন্যথায় ফলাফল False। AndAlso- অর্থাৎ এবং যদিও। উভয় এক্সপ্রেশন সমান থাকলে ফলাফল True। OrElse - উভয়টির যে কোনো একটি সত্য হলে ফলাফল True। IsFalse -এটা বের করে কোনো একটি এক্সপ্রেশন মিথ্যা কি না। IsTrue -এটা নির্ধারণ করে কোনো একটি এক্সপ্রেশন সত্য কি না।

* বিট শিফ্‌ট অপারেটর [Bit Shift Operators] - এটা হচ্ছে বাইনারি পার্থক্য নির্ণয় পদ্ধতি। যেমন: A = 60 ও B = 13 হলে, বাইনারিতে তা হবে A = 0011 1100 এবং B = 0000 1101।

* এসাইনমেন্ট প্রতীক [Assignment Operators] -এগুলো হলো: =, += [Add AND- উদাহরণ: চ += ক, ক এর মান রেখে তাতে যোগ হবে চ এর মান। এটা চ = চ + ক এর সমান।]। -= [Subtract AND -আগেরটির বিপরীত ক্রিয়া]- চ = চ - ক এর সমান, । *= [Multiply AND - C *= A হলো C = C * A]। /= [Divide AND -C /= A হলো C = C/ A এর অনুরূপ]। \= [C \= A হলো C = C \A এর অনুরূপ]। ^= [C^=A হলো C = C ^ A এর অনুরূপ]।

১৪. সিদ্ধান্ত নেওয়া: Decision Making

প্রোগ্রামিংয়ে সিদ্ধান্ত নেওয়া একটি অবশ্যকরণীয় কাজ। ব্যবহারযোগ্য এমন কোনো এ্যাপলিকেশন হয়তো পাওয়া যাবে না যার কোডে সিদ্ধান্ত নেওয়ার এক বা একাধিক ব্লক নেই। টেস্টসহ সিদ্ধান্ত নেওয়া প্রোগ্রামিংয়ে একান্ত জরুরী একটি ব্যাপার। নিচের ফ্লো চার্টটি অধিকাংশ প্রোগ্রামিং ভাষায় মডেল হিসেবে ব্যবহৃত হয়।



ভিজুয়্যাল বেসিক.নেট ভাষার মধ্যে কিছু বিশেষ কিওয়ার্ড আছে যেগুলো ব্যবহৃত হয় সিদ্ধান্ত নিতে। যেমন:
১. If ... Then স্টেটমেন্টস্। ২. If ... Then .... Else স্টেটমেন্টস্ ৩. Nested If স্টেটমেন্টস্। ৪. Select Case স্টেটমেন্টস্। ৫. Nested Select Case স্টেটমেন্টস্। এগুলোর সব ক'টিই আমরা ব্যবহার করেছি এ গ্রন্থের বিভিন্ন উদহারণ এ্যাপলিকেশনে। এরপরও প্রতিটির ওপর একেকটি করে উদাহরণ দিচ্ছি সবার সুবিধার্থে।

১. a যদি ২০ থেকে কম বা সমান হয় তাহলে c= c+1 ।

```
If (a <= 20) Then
    c= c+1
End If
```

২. a যদি ২০ থেকে কম বা সমান হয় না তাহলে msgBox("a is larger then 20") ।

```
If (a <= 20) Then
    c= c+1
Else
    msgBox("a is larger then 20")
End If
```

৩. a যদি ১০০ হয় তাহলে b যদি ২০০ হয় তবেই পরের লাইন লিখা হবে: "Value of a is 100 and b is 200" ।

```
If (a = 100) Then
      ' if condition is true then check the following
      If (b = 200) Then
         ' if condition is true then print the following
         Console.WriteLine("Value of a is 100 and b is 200")
      End If
End If
```

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

৪. নিচের কোডে গ্রেড A, B, C, D ও F পর্যন্ত সিলেক্ট কেইস ব্যবহৃত হয়েছে।

```
Sub Main()
    'local variable definition
    Dim grade As Char
    grade = "B"
    Select grade
        Case "A"
            Console.WriteLine("Excellent!")
        Case "B", "C"
            Console.WriteLine("Well done")
        Case "D"
            Console.WriteLine("You passed")
        Case "F"
            Console.WriteLine("Better try again")
        Case Else
            Console.WriteLine("Invalid grade")
    End Select
    Console.WriteLine("Your grade is  {0}", grade)
    Console.ReadLine()
End Sub
```

৫. নিচের কোডে নেস্টেড সিলেক্ট কেইস স্টেটমেন্ট ব্যবহৃত হয়েছে।

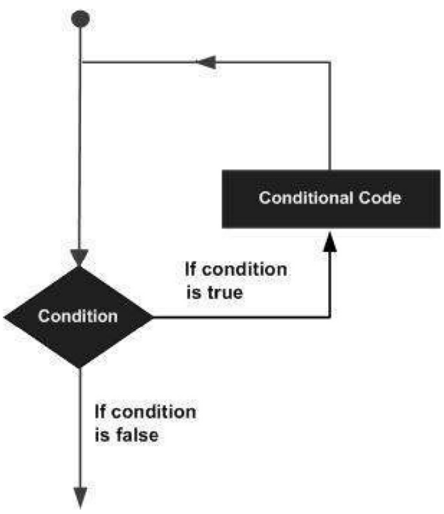
```
Module decisions
   Sub Main()
    'local variable definition
    Dim a As Integer = 100
    Dim b As Integer = 200
    Select a
        Case 100
            Console.WriteLine("This is part of outer case ")
            Select Case b
                Case 200
                    Console.WriteLine("This is part of inner case ")
            End Select
    End Select
    Console.WriteLine("Exact value of a is : {0}", a)
    Console.WriteLine("Exact value of b is : {0}", b)
    Console.ReadLine()
   End Sub
End Module
```

বায়ের প্রোগ্রাম রানের ফলাফল:

```
This is part of outer case
This is part of inner case
Exact value of a is : 100
Exact value of b is : 200
```

১৫. ভিজুয়্যাল বেসিক.নেট লুপ: VB.Net Loops

অধিকাংশ প্রোগ্রামিং ভাষায় লুপের লজিক ডানের চিত্রের মতো ব্যবহৃত হয়। প্রোগ্রামে লুপও একটি গুরুত্বপূর্ণ বিষয়। সুতরাং এ সম্পর্কে ভালো জ্ঞান থাকা আবশ্যক। সাধারণত, প্রোগ্রামের স্টেটমেন্ট একটার পর আরেকটি এক্সেকিউট হয়। সুতরাং কোথায় কোন্ কোড লাইন থাকবে তা-ও গুরুত্বহ। অপরদিকে লুপিংয়ের কাজ হলো কোনো শর্ত পূরণ না হওয়া পর্যন্ত লুপের ভেতর ঘুরে ঘুরে এক্সেকিউশন চলতে থাকবে। এ গ্রন্থের বিভিন্ন উদাহরণে অনেক লুপ আমরা দেখেছি।



১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সর্বমোট ৬ ধরনের লুপিং তার ভাষার মধ্যে সম্পৃক্ত করেছে। আপনার কাজের জন্য যেটি উপযোগী সেটিই ব্যবহার করবেন। নিচের টেবিলে এ ৬টি লুপ কাঠামোর ওপর তথ্যাদি তুলে ধরছি।

Loop Type	Description
Do Loop	It repeats the enclosed block of statements while a Boolean condition is True or until the condition becomes True. It could be terminated at any time with the Exit Do statement.
For...Next	It repeats a group of statements a specified number of times and a loop index counts the number of loop iterations as the loop executes.
For Each...Next	It repeats a group of statements for each element in a collection. This loop is used for accessing and manipulating all elements in an array or a VB.Net collection.
While... End While	It executes a series of statements as long as a given condition is True.
With... End With	It is not exactly a looping construct. It executes a series of statements that repeatedly refer to a single object or structure.
Nested loops	You can use one or more loops inside any another While, For or Do loop.

লুপ ব্যবহার পদ্ধতি: উপরের ৬টি লুপ কিভাবে কোডিং করতে হয় তার উদাহরণ এবার ততুলে ধরছি।

Do Loop

```
Module loops
  Sub Main()
    ' local variable definition
    Dim a As Integer = 10
    'do loop execution
    Do
       Console.WriteLine("value of a:
{0}", a)
       a = a + 1
    Loop While (a < 20)
    Console.ReadLine()
  End Sub
End Module
======
Result:
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
```

For...Next

```
Module loops
  Sub Main()
    Dim a As Byte
    ' for loop execution
    For a = 10 To 20
       Console.WriteLine("value of a:
{0}", a)
    Next
    Console.ReadLine()
  End Sub
End Module
======
Result:
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
value of a: 20
```

For Each...Next

```
Module loops
  Sub Main()
    Dim anArray() As Integer = {1, 3,
5, 7, 9}
    Dim arrayItem As Integer
    'displaying the values
    For Each arrayItem In anArray
       Console.WriteLine(arrayItem)
    Next
    Console.ReadLine()
  End Sub
End Module
======
Result:
1
3
5
7
9
```

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

While ... End While

```
Module loops
   Sub Main()
      Dim a As Integer = 10
      ' while loop execution '
      While a < 20
          Console.WriteLine("value of a: {0}", a)
          a = a + 1
      End While
      Console.ReadLine()
   End Sub
End Module
```

======

Result:
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19

Nested Loop

```
Module loops
  Sub Main()
    ' local variable definition
    Dim i, j As Integer
    For i = 2 To 100
      For j = 2 To i
        ' if factor found, not prime
        If ((i Mod j) = 0) Then
          Exit For
        End If
      Next j
      If (j > (i \ j)) Then
        Console.WriteLine("{0} is prime", i)
      End If
    Next i
    Console.ReadLine()
  End Sub
End Module
```

Result:
2 is prime
3 is prime
5 is prime
7 is prime
11 is prime
13 is prime
17 is prime
19 is prime
23 is prime
29 is prime
31 is prime
37 is prime
41 is prime
43 is prime
47 is prime
53 is prime
59 is prime
61 is prime
67 is prime
71 is prime
73 is prime
79 is prime
83 is prime
89 is prime
97 is prime

With...End With

```
Module loops
  Public Class Book
    Public Property Name As String
    Public Property Author As String
    Public Property Subject As String
  End Class
  Sub Main()
    Dim aBook As New Book
    With aBook
      .Name = "VB.Net Programming"
      .Author = "Engineer Azizul Bari"
      .Subject = "IT"
    End With
    With aBook
      Console.WriteLine(.Name)
      Console.WriteLine(.Author)
      Console.WriteLine(.Subject)
    End With
    Console.ReadLine()
  End Sub
End Module
```

Result:
VB.Net Programming
Engineer Azizul Bari
IT

এ ছ'টি প্রোগ্রাম ইচ্ছে হলে রান করে দেখতে পারেন। মনে রাখবেন লুপ কন্ট্রোল স্টেটমেন্ট কোড চলার সাধারণ পরম্পরাকে পরিবর্তন করে। লুপের ভেতর থেকে বের হওয়ার জন্য তিনটি কিওয়ার্ড ব্যবহার করা যায়। যেমন: ক. Exit - লুপ বা সিলেক্ট স্টেটমেন্ট থেকে বের হয়ে পরবর্তী কোডগুলো এক্সেকিউট হবে। খ. Continue - এটা বাকি কোড এক্সেকিউট না করে লুপের প্রথমে ফিরে যায়। গ. GoTo -এটা আজকাল তেমন বেশি ব্যবহৃত হয় না। কিওয়ার্ডের পরে একটি নাম থাকতে হবে। একে লেবেল বলে। এই নামটি যেখানে আছে সেখানে জাম্প মারবে।

```
Module loops
  Sub Main()
    ' local variable definition
    Dim a As Integer = 10
    ' while loop execution '
    While (a < 20)
      Console.WriteLine("value of a: {0}", a)
      a = a + 1
      If (a > 15) Then
        'terminate the loop using exit statement
        Exit While
      End If
    End While
    Console.ReadLine()
  End Sub
End Module
```

বায়ের কোড রানের ফলাফল:
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15

লক্ষ করুন, Exit While লাইনটি কার্যকর হবে যেই মুহূর্তে a এর মান ১৫ থেকে বেশি তথা ১৬ হবে।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

১৫. ভিজুয়্যাল বেসিক.নেটে স্ট্রিং নিয়ে কাজ: VB.Net Strings Manipulation

ভিজুয়্যাল বেসিক.নেটে String শব্দটি বিশেষ স্থান দখল করে আছে। স্ট্রিং মানে কিবোর্ডের যে কোনো এক বা একাধিক ক্যারাক্টারকে বুঝায়। String একটি টাইপও বটে। যেমন: আমরা বলতে পারি Dim Name as String। এই ডিক্লারেশনে String টাইপ। স্ট্রিং নিয়ে অনেক কাজ করা যায়। নিচে কছু উদাহরণ দেওয়া হলো।

১. স্ট্রিং বস্তু সৃষ্টির পদ্ধতি ৫টি। যথা: (ক) স্ট্রিং ভেরিয়েবলে স্ট্রিং লিটারেল এসাইন করা। যেমন: Dim Name as String; Name = "E. A. Bari"। (খ) স্ট্রিং কন্সট্রাক্টারের ব্যবহার। যেমন: Dim letters As Char() = {"H", "e", "l", "l", "o"}; greetings = New String(letters)। (গ) স্ট্রিং যোগ [+] অপরেটর ব্যবহার। যেমন: Dim fName, sName, FullName as String; fName = "Muhammad"; sName = "Hassan"; FulName = fName + " " + sName। (ঘ) মেথড কল করে স্ট্রিং রিটার্ন করা। যেমন: Dim sarray() As String = {"Hello", "From", "Subidbazar", "Point."}; Dim message As String = String.Join(" ", sarray)। এখন message = Hello From Subidbazar Point.। (ঙ) মেথড ফরম্যাট করে। যেমন: Dim waiting As DateTime = New DateTime(2017, 1, 12, 17, 58, 1); Dim chat As String = String.Format("Message sent at {0:t} on {0:D}", waiting); Console.WriteLine("Message: {0}", chat)। ফলাফল: Message: Message sent at 5:58 PM on Thursday, January 12, 2017 ।

স্ট্রিং অবেক্টের অনেক ফাংশন ও মেথড আছে। ওগুলোর ওপর এখানে আর কিছু বলে কলেবর বৃদ্ধি করছি না। শুধুমাত্র নামগুলো এখানে উল্লেখ করবো। প্রোগ্রাম করাকালে আপনি ইন্টারনেট বা স্টুডিও এর Help থেকে সহজেই জানতে ও ব্যবহার করতে পারবেন।

1. String.Compare(str1, str2) [result 0 if both are equal]
2. str1.contains(str2) - if str1 contains str2 then result is True, otherwise False
3. string.Substring (position as integer)gets the sring's position to end. ex. "This Book".Substring(5) -result Book.
4. String.Join (vbCrLf, Sring Value) - joins string.

Example:

```
Dim strarray As String() = {"Down the way where the nights are grey",
                 "And the sun shines daily on the mountain top",
                  "I took a trip on a sailing ship",
                 "And when I reached Jamaica",
                 "I made a stop"}
Dim str As String = String.Join(vbCrLf, strarray)
Result:   Down the way where the nights are grey
          And the sun shines daily on the mountain top
          I took a trip on a sailing ship
          And when I reached Jamaica
          I made a stop
```

১৬. তারিখ নিয়ে কাজ: Working with Date

ভিজুয়্যাল বেসিক.নেট বেশ কয়েকটি Date ফাংশন তৈরি করেছে আমাদের সুবিধার্থে। এর ক'টির ব্যবহার ডানের কোড অংশে দেখুন। ফলাফল নিম্নরূপ।

```
12/16/2012 12:00:00 PM
12/16/2012 12:00:52 PM
1/18/2017 9:55:18 PM [Local Time BD]
1/18/2017 3:55:18 PM [GMT]
1/18/2017 12:00:00 AM
Current Time: 9:55:18 PM
```

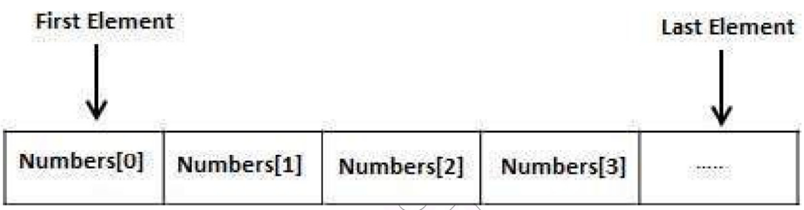
```
Module Module1
  Sub Main()
    'DateTime constructor: parameters year, month,
day, hour, min, sec
    Dim date1 As New Date(2012, 12, 16, 12, 0, 0)
    'initializes a new DateTime value
    Dim date2 As Date = #12/16/2012 12:00:52 AM#
    'using properties
    Dim date3 As Date = Date.Now
    Dim date4 As Date = Date.UtcNow
    Dim date5 As Date = Date.Today
    Console.WriteLine(date1)
    Console.WriteLine(date2)
    Console.WriteLine(date3)
    Console.WriteLine(date4)
    Console.WriteLine(date5)
    Console.Write("Current Time: ")
    Console.WriteLine(Now.ToLongTimeString)
    Console.ReadKey()
  End Sub
End Module
```

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

১৭. ভিজুয়্যাল বেসিক.নেট এ্যারে: VB.Net - Arrays

এ্যারে ভেরিয়েবলে তথ্য গুচ্ছ একত্রে সংরক্ষণ করা যায়। এ্যারের মধ্যে থাকে ০-ইনডেক্স ইলেমেন্ট। নিচের চকটি দেখুন।



আমরা Dim সেস্টমেন্ট দ্বারা এ্যারে সৃষ্টি করি। যেমন:

```
Dim intData(30)  ' an array of 31 elements
Dim strData(20) As String' an array of 21 strings
Dim twoDarray(10, 20) As Integer 'a two
dimensional array of integers
Dim ranges(10, 100) 'two dimensional array
```

ডিক্লারেশনের সময় এ্যারে ইনিশিয়েলিজও করা যায়। যেমন:

```
Dim intData() As Integer = {12, 16, 20, 24, 28, 32} 'contains 0-5 indexed integer array elements
Dim Mnames() As String = {"Boishak", "Joysta", _
"Ashar", "Srabon", "Vadra"} 'contains 0-4 indexed string array elements
Dim miscData() As Object = {"Hello World", 12d, 16ui, "A"c} 'contains 0-3 indexed mixed types array elements
```

যে কোনো এ্যারে ভেরিয়েবলে সংরক্ষিত ডাটা ইলেমেন্ট ইনডেক্সট ব্যবহার করে বের করে আনা যায়। যেমন উপরের প্রথম এ্যারের চতুর্থ সংখ্যাটি পেতে আমরা লিখবো: intData(3) ফলাফল হবে ২৪। অনুরূপ: Mnames(0) এর ফলাফল হবে Boishak।

ডাইনামিক এ্যারে: Dynamic Arrays: এর অর্থ প্রোগ্রাম চলাকালে প্রয়োজনে আমরা এ্যারে ডাইমেনশন করতে পারি ও পুনঃডাইমেনশনও করতে পারি। এজন্য আমরা ব্যবহার করি ReDim কিওয়ার্ডটি। এটা ঘোষণভর Syntax হলো:

ReDim [Preserve] arrayname(subscripts)

লক্ষ করুন, অপশন্যাল কিওয়ার্ড Preserve ব্যবহার করলে এ্যারের পূর্বের ইলেমেনন্টগুলোও অপরিবর্তিত থাকবে। যেমন:

```
Dim ExamMarks() as Integer 'array declaration
ReDim ExamMarks(5) 'now the array has 0-6 indexed integer elements.
ReDim Preserve ExamMarks(10) 'now the array has 0-11 indexed integer elements.
```

একাধিক বিস্তৃতির এ্যারে: Multi-Dimensional Arrays: এ্যারে ভেরিয়েবল একাধিক বিস্তৃতিসম্পন্নও হতে পারে। যেমন আমরা ডিক্লায়ার করতে পারি: DIm twoDStringArray(10, 20) As String। এটা ২-বিস্তৃতির এ্যারে। ৩-বিস্তৃতিসম্পন্ন এ্যারে ডিক্লারেশন হলো: Dim threeDIntArray(10, 10, 10) As Integer। নিচে ২-বিস্তৃতির একটি এ্যারে কোড ব্লক দেখুন।

```
Module arrayApl
  Sub Main()
    ' an array with 5 rows and 2 columns
    Dim a(,) As Integer = {{0, 0}, {1, 2}, {2, 4}, {3, 6}, {4, 8}}
    Dim i, j As Integer
    ' output each array element's value '
    For i = 0 To 4
      For j = 0 To 1
        Console.WriteLine("a[{0},{1}] = {2}", i, j, a(i, j))
      Next j
    Next i
    Console.ReadKey()
  End Sub
End Module
```

বায়ের কোড থেকে প্রাপ্ত ফলাফল:

```
a[0,0]: 0
a[0,1]: 0
a[1,0]: 1
a[1,1]: 2
a[2,0]: 2
a[2,1]: 4
a[3,0]: 3
a[3,1]: 6
a[4,0]: 4
a[4,1]: 8
```

ভিজুয়্যাল বেসিক.নেট এ্যারে ক্লাসের বেশ ক'টি মেথড তৈরি করে দিয়েছে। আমরা এগুলো প্রয়োজনে কাজে লাগাতে

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

পারি। এসব মেথডের মধ্যে উল্লেখযোগ্য ক'টি হলো:
Clear, Copy, CopyTo, GetLength, GetLongLength, GetLowerBound, GetType, GetUpperBound, GetValue, IndexOf, Reverse, SetValue, Sort, ToString.

এবার একটা উদাহরণ দেখে নিই। নিচের মডিউলে এ্যারে ক্লাসের কিছু মেথড ব্যবহারের নিয়ম তুলে ধরা হয়েছে।

```
Module arrayApl
  Sub Main()
    Dim list As Integer() = {34, 72, 13, 44, 25, 30, 10}
    Dim temp As Integer() = list
    Dim i As Integer
    Console.Write("Original Array: ")
    For Each i In list
       Console.Write("{0} ", i)
    Next i
    Console.WriteLine()
    ' reverse the array
    Array.Reverse(temp)
    Console.Write("Reversed Array: ")
    For Each i In temp
       Console.Write("{0} ", i)
    Next i
    Console.WriteLine()
    'sort the array
    Array.Sort(list)
    Console.Write("Sorted Array: ")
    For Each i In list
       Console.Write("{0} ", i)
    Next i
    Console.WriteLine()
    Console.ReadKey()
  End Sub
End Module
```

বায়ের প্রোগ্রাম কম্পাইল করে রান করলে যে ফলাফল পাওয়া যাবে তাহলো:
Original Array: 34 72 13 44 25 30 10
Reversed Array: 10 30 25 44 13 72 34
Sorted Array: 10 13 25 30 34 44 72

লক্ষ করুন, উপরের প্রথম লিস্ট হলো মূল এ্যারে। দ্বিতীয়টি উল্টো এ্যারে এবং তৃতীয়টি ছোট থেকে বড় সর্টকরা এ্যারে।

১৮. ফাংশন: VB.Net - Functions

ফাংশনের ওপর বেশ কিছু আলোচনা, ব্যাখ্যা ও ব্যবহার উদাহরণ আমরা এই গ্রন্থে তুলে ধরেছি। এরপরও আরো কিছু জানার আছে। ফাংশন মূলত সাবরুটিন প্রসিডোর এর মতো কাজ করে। তবে ফাংশন থেকে আমরা পেতে পারি মান- যা এটা কলিং সাবে রিটার্ন করে। সুতরাং বলা যায়, ফাংশন একটি ভ্যালু রিটার্ন করে কিন্তু সাবরুটিন তা করে না।

ফাংশন ডিফাইন করা নিয়ম হলো:

```
[Modifiers] Function FunctionName [(ParameterList)] As ReturnType
   [Statements]
End Function
```

এখানে, Modifiers অপশন্যাল এবং তা নির্দিষ্ট করে ফাংশন প্রাপ্তির লেবেল। এগুলো হলো: Public, Private, Protected, Friend, Protected Friend । FunctionName জরুরি। ParameterList অপশন্যাল। সবশেষে ReturnType নির্দিষ্ট করে কোন্ ধরনের ডাটা ফাংশন রিটার্ন করবে।

ফাংশনের বিভিন্ন দিক বুঝতে আমরা কয়েকটি উদাহরণ দেখতে পারি। পরের পৃষ্ঠায় এসব উদাহরণ তুলে ধরেছি। শিক্ষার্থীরা চাইলে এসব ফাংশনের ফলাফল প্রজেক্ট তৈরি করে পরীক্ষা করতে পারেন। বাস্তবে তা করাটাই উত্তম।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

ফাংশন উদাহরণ -১: এটা দু'টি ইন্টেজার প্যারামিটার নেয় এবং রিটার্ন করে এদের মধ্যে বড়ো সংখ্যাটি।

```
Function FindMax(ByVal num1 As Integer, ByVal num2 As Integer) As Integer
  ' local variable declaration */
  Dim result As Integer
  If (num1 > num2) Then
     result = num1
  Else
     result = num2
  End If
  FindMax = result
End Function
```

উদাহরণ -২: ভিজুয়্যাল বেসিক.নেটে ফাংশন ভ্যালু বা মান রিটার্ন করে দু'টি পদ্ধতিতে। প্রথমটি হলো Return কিওয়ার্ড দ্বারা আর দ্বিতীয়টি হলো ফাংশনের নামের দ্বারা। উপরের ফংশনে ব্যবহৃত হয়েছে দ্বিতীয় পদ্ধতি। নিচের ফাংশন প্রথম পদ্ধতিতে ভ্যালু রিটার্ন করবে। এটা সংখ্যার ফ্যাক্টোরিয়েল মান বের করে।

```
Module myfunctions
  Function factorial(ByVal num As Integer) As Integer
    ' local variable declaration */
    Dim result As Integer
    If (num = 1) Then
       Return 1
    Else
       result = factorial(num - 1) * num
       Return result
    End If
  End Function
```

উদাহরণ - ৩: কেনো কোনো সময় এমন হয় যে, ফাংশনের মধ্যে ক'টি প্যারামিটার হবে তা নির্দিষ্ট করে বলা যায় না। এমতাবস্থায় কি করবো? ভিজুয়্যাল বেসিক.নেট এরূপ ক্ষেত্রে একটি কিওয়ার্ড দিয়েছে। এর নাম ParamArray। ফাংশন ডিক্লারেশনে এটি ব্যবহার করে অনির্দিষ্ট ইলেমেন্টবিশিষ্ট এ্যারে ব্যবহার করতে পারি ফাংশনের প্যারামিটার হিসাবে। দেখুন এই উদাহরণটি।

```
Module myparamfunc
  Function AddElements(ParamArray arr As Integer()) As Integer
    Dim sum As Integer = 0
    Dim i As Integer = 0
    For Each i In arr
       sum += i
    Next i
    Return sum
  End Function
  Sub Main()
    Dim sum As Integer
    sum = AddElements(512, 720, 250, 567, 889)
    Console.WriteLine("The sum is: {0}", sum)
    Console.ReadLine()
  End Sub
End Module
```

ফাংশনের ব্যাখ্যা: আমরা এখানে ParamArray কিওয়ার্ড দ্বারা একটি অনির্দিষ্ট ইলেমেন্টবিশিষ্ট প্যারামিটার তৈরি করেছি। ঠিক ক'টি ইলেমেন্ট থাকবে তা কলিং সাব দ্বারা জানিয়েছি। বলেছি: sum = AddElements(512, 720, 250, 567, 889)। সুতরাং প্যারামিটার হয়েছে মোট ৫টি।

ফাংশনের ফলাফল হলো এই ৫টি সংখ্যার যোগফল:

The sum is: 2938

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

উদাহরণ -৪: সময় সময় এ্যারেকে ফাংশনের আর্গুমেন্ট হিসেবেও পাস করা যায়। যেমন:

```
Module arrayParameter
  Function getAverage(ByVal arr As Integer(), ByVal size As Integer) As Double
    'local variables
     Dim i As Integer
     Dim avg As Double
     Dim sum As Integer = 0
     For i = 0 To size - 1
        sum += arr(i)
     Next i
     avg = sum / size
     Return avg
  End Function
  Sub Main()
     ' an int array with 5 elements '
     Dim balance As Integer() = {1000, 2, 3, 17, 50}
     Dim avg As Double
     'pass pointer to the array as an argument
     avg = getAverage(balance, 5)
     ' output the returned value '
     Console.WriteLine("Average value is: {0} ", avg)
     Console.ReadLine()
  End Sub
End Module
```

ফাংশনের ব্যাখ্যা: ব্যবহারকারী ক'টি সংখ্যার গড় বের করবেন তা আমাদের জানা নেই। সুতরাং আমরা একটি ফাংশন তৈরি করলাম যা মূলত ফ্লেক্সিবল। বায়ের কলিং সাবে আমরা বলেছি balance নামক Integer এ্যারেতে মোট ৫টি সংখ্যা আছে। আমরা এগুলোর গড় ফাংশন থেকে বের করার নির্দেশ দিয়েছি। লক্ষ করুন নির্দেশটি হলো এই লাইন:

avg = getAverage(balance, 5)

ফাংশনের প্রথম প্যারামিটার একটি অনির্দিষ্ট এ্যারে। আর দ্বিতীয়টি এ্যারের সাইজ। উপরের লাইন দ্বারা এ্যারে ও সাইজ উভয়টি নির্দিষ্ট করে ফাংশন কল করেছি। আর সঠিক ফালাফলও পেয়েছি:

Average value is: 214.4

১৮. সাব প্রসিডোর: VB.Net - Sub Procedure

এই গ্রন্থের সর্বত্র আমরা সাব প্রসিডোর ব্যবহার করে আসছি। সুতরাং প্রসিডোর সম্পর্কে শিক্ষাথীরা পূর্ণাঙ্গরূপে অবগত আছেন। এরপরও দু'একটি বিষয় আমাদের জানার আছে। এখন ওসব ব্যাপার নিয়ে সামান্য আলোচনা করবো। প্রথমে দেখে নিই সাব প্রসিডোরের সিনট্যাক্সট।

```
[Modifiers] Sub SubName [(ParameterList)]
    [Statements]
End Sub
```

অপশন্যাল Modifiers হতে পারে: Public, Private, Protected, Friend, Protected Friend । একটি উদাহরণ দেখে নিই।

```
Module mysub
  Sub CalculatePay(ByRef hours As Double, ByRef wage As Decimal)
    'local variable declaration
    Dim pay As Double
    pay = hours * wage
    Console.WriteLine("Total Pay: {0:C}", pay)
  End Sub
  Sub Main()
    'calling the CalculatePay Sub Procedure
    CalculatePay(25, 10)
    CalculatePay(40, 20)
    CalculatePay(30, 27.5)
    Console.ReadLine()
  End Sub
End Module
```

সাবের ব্যাখ্য ও ফলাফল: CalculatePay সাবরুটিনে আছে hours এবং wage নামক দু'টি প্যারামিটার। সাবই ফলাফল দিয়েছে। কলিং নাম [Sub Main] থেকে আমরা তিনবার সাবরুটিন কল করেছি। আর ফলাফল যা পেয়েছি তাহলো:

Total Pay: $250.00
Total Pay: $800.00
Total Pay: $825.00

ভিজুয়্যাল বেসিক.নেটে ফাংশন ও সাব প্রসিডোরের আর্গুমেন্ট বা প্যারামিটার দু'ভাবে ডিফাইন করা যায়। এগুলো হলো: ByValue এবং ByRef। প্রথমটি হলো ডিফোল্ট। প্রথমটির অর্থ আর্গুমেন্ট অপরিবর্তিত থাকবে, দ্বিতীয়টির অর্থ, প্রয়োজনে ফাংশন বা সাবরুটিন আর্গুমেন্টের মান রদবদল করতে পারবে।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

১৯. ক্লাস ও অবজেক্ট: Classes & Objects

ক্লাস ডেফিনিশন মানে একটি ডাটা টাইপের জন্য ব্লুপ্রিন্ট তৈরি করা। অবজেক্ট হলো ক্লাসের ইন্সস্ট্যান্ট। ক্লাসের ভেরতে যেসব মেথড ও ভেরিয়েবল থাকে ওগুলোকে বলে ক্লাসের মেমবার। ক্লাসের একটি উদাহরণ দেখে নিই।

```
Class Box
    Public length As Double   ' Length of a box
    Public breadth As Double   ' Breadth of a box
    Public height As Double   ' Height of a box
End Class
```

উপরের ক্লাসে একটি বাক্সের ডেফিনিশন এসছে। এর ভেতর তিনটি সদস্য আছে: length, breadth and height । এটাকে কাজে লাগাতে আমরা প্রথমে ইন্সস্ট্যান্ট ডিক্লারেশন করে থাকি। যথা:

```
Dim Box1 As Box = New Box()       ' Declare Box1 of type Box
Dim Box2 As Box = New Box()       ' Declare Box2 of type Box
```

এখন আমরা লিখতে পারি:

```
Box1.height = 5.0
Box1.length = 6.0
Box1.breadth = 7.0
' box 2 specification
Box2.height = 10.0
Box2.length = 12.0
Box2.breadth = 13.0
```

উপরোক্ত ভ্যালুগুলো নিয়ে ক্যালকুলেশনের জন্য ডিক্লায়ার করতে পারি আরেকটি ভেরিয়েবল। যেমন:

```
Dim volume As Double = 0.0
```

চাইলে আমরা ক্লাসের ভেতরই উক্ত বাক্সের তিনটি ডাইমেনশন ডিফাইন ও ক্যালকুলেট করতে পারি। যেমন:

```
Class Box
    Public length As Double   ' Length of a box
    Public breadth As Double   ' Breadth of a box
    Public height As Double   ' Height of a box
    Public Sub setLength(ByVal len As Double)
        length = len
    End Sub
    Public Sub setBreadth(ByVal bre As Double)
        breadth = bre
    End Sub
    Public Sub setHeight(ByVal hei As Double)
        height = hei
    End Sub
    Public Function getVolume() As Double
        Return length * breadth * height
    End Function
  End Class
```

লক্ষ করুন, আমরা প্রথম ক্লাস ডেফিনিশনে শুধুমাত্র বাক্সের তিনটি ডাইমেনশনের নাম টাইপসহ ডিক্লায়ার করেছি। কিন্তু উপরের ক্লাসে তিনটি ডাইমেনশন সেট ও ভলিউম বের করার ফাংশনও ডিফাইন করেছি। এর অর্থ হলো কলিং সাবের জন্য কাজটি সহজ করে দিলাম। এখন কলিং সাবে আমরা লিখতে পরি:

```
Dim Box1 As Box = New Box()       ' Declare Box1 of type Box
Dim volume As Double = 0.0    ' Store the volume of a box here
```

এরপর আমরা সেট করতে পারি মান: [পরের পৃষ্ঠায় দেখুন]

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

```
' box 1 specification
Box1.setLength(6.0)
Box1.setBreadth(7.0)
Box1.setHeight(5.0)
' volume of box 1
volume = Box1.getVolume() 'function call within the Box class
```

বর্ণিত উপায়ে ক্লাস ডিক্লারেশনকে বলে Member Functions and Encapsulation পদ্ধতি।

২০. এক্সসেপশন বা এ্যারোর নিয়ন্ত্রণ: Exception or Error Handling

কোডিং করতে যেয়ে ভুল হবেই। সামান্য ভুলের কারণে পুরো এ্যাপলিকেশনই নষ্ট হয়ে যেতে পারে। এমনকি রানকালে কম্পিউটার ক্রাশও করতে পারে। অনাকাঙ্ক্ষিত এই অবস্থা থেকে নিষ্কৃতির কিছু উপায় ভিজুয়্যাল বেসিক.নেট তৈরি করেছে। এসব উপায় অবলম্বনের নামই হলো **Exception or Error Handling**।

সাধারণত কোডের যে অংশে ভুল আছে সেখান থেকে প্রোগ্রাম ফ্লো অন্যত্র ট্রান্সফার করে নেওয়া হয়ে থাকে। ভিজুয়্যাল বেসিক.নেট ভুল সংশোধনের জন্য চারটি কিওয়ার্ড দিয়েছে: **Try, Catch, Finally** and **Throw**। আরেকটি কিওয়ার্ড আছে। আজকাল এটি খুব বেশি ব্যবহৃত হয় না। যদিও এক সময় এটা খুব জনপ্রিয় ছিলো। এই কিওয়ার্ডটি হলো: GoTo [label]। অর্থাৎ নির্দিষ্ট লেবেলে জাম্প করার নির্দেশ। যা হোক প্রথমোক্ত চারটি কিওয়ার্ড কিভাবে কাজ করে একটু দেখে নিই।

```
Module exceptionProg
  Sub division(ByVal num1 As Integer, ByVal num2 As Integer)
    Dim result As Integer
    Try
      result = num1 \ num2
    Catch e As DivideByZeroException
      Console.WriteLine("Exception caught: {0}", e)
    Finally
      Console.WriteLine("Result: {0}", result)
    End Try
  End Sub
  Sub Main()
    division(25, 0)
    Console.ReadKey()
  End Sub
End Module
```

বায়ের সাব ও ফলাফল: এখানে Try ... End Try ব্লকের ভেতর ভুল ধরার ব্যবস্থা করা হয়েছে। মনে করুন ০ দ্বারা ভাগের চেষ্টা করা হলো, যেমনটি Sub Main-এ করা হয়েছে। যেহেতু এটা ভুল তাই উক্ত ব্লকে এটা ধরা পড়বে ও প্রোগ্রাম রিপোর্ট দেবে। বায়ের কোড রানের ফলাফল হবে:

Exception caught: System.DivideByZeroException: Attempted to divide by zero.
at ...
Result: 0

প্রোগ্রামার নিজেও ভুল ধরার ডেফিনিশন তৈরি করতে পারেন। যেমন:

```
Public Class TempIsZeroException : Inherits ApplicationException
    Public Sub New(ByVal message As String)
        MyBase.New(message)
    End Sub
End Class
Public Class Temperature
    Dim temperature As Integer = 0
    Sub showTemp()
        If (temperature = 0) Then
            Throw (New TempIsZeroException("Zero Temperature found"))
        Else
            Console.WriteLine("Temperature: {0}", temperature)
        End If
    End Sub
End Class
```

বায়ের প্রথম ক্লাস ডিক্লারেশনের সময় ApplicationException নামক সিস্টেম ক্লাস ব্যবহৃত হয়েছে। এরপর ক্লাসে কিওয়ার্ড দ্বারা এক্সেস এক্সপ্রেশনের মেসেজটি ডিফাইন করা হয়েছে। সুতরাং ভুল হলে লেখা আসবে: TempIsZeroException: Zero Temperature found

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

২১. ফাইল নিয়ে কাজ: File Handling

ফাইল কি? ফাইল হচ্ছে একগুচ্ছ তথ্য যা ডিস্কের মধ্যে সংরক্ষিত থাকে। ফাইল থাকে ডাইরেক্টরিতে যার একটি নির্দিষ্ট প্যাথ বা রাস্তা আছে। যে মুহূর্তে কোনো ফাইল খুলবে পড়া ও লেখার জন্য, তখনই এটা হয়ে যায় ফাইল-স্ট্রীম। স্ট্রীম কি? এটা মূলত পরম্পরা তথ্যের বাইটস যোগাযোগ রাস্তা ধরে চলমান থাকে। দু'টি মূল স্ট্রীম আছে: ইনপুট [প্রবেশকরণ] ও আউটপুট [বেরকরণ]। ইনপুট স্ট্রিম ব্যবহৃত হয় ফাইল থেকে ডাটা পড়ার জন্য। আর আউটপুট স্ট্রিম দ্বারা ফাইলের মধ্যে ডাটা লিখা হয়।

ভিজুয়্যাল বেসিক.নেট সিস্টেম আইও [System.IO namespace] নেইমস্পেস তৈরি করেছে ফাইল নিয়ে কাজ করার জন্য। যেমন: ফাইল সৃষ্টি, ধ্বংস, খোলা, বন্ধ করা, পড়া, লেখা, সংরক্ষণ করা ইত্যাদি। নিচের টেবিলে System.IO namespace-এর সচরাচর ব্যবহৃত কয়েকটি ক্লাস ব্যাখ্যাসহ তুলে ধরেছি। পরে কিছু উদাহরণ কোডও লিপিবদ্ধ করেছি।

I/O Class	Description
BinaryReader	Reads primitive data from a binary stream.
BinaryWriter	Writes primitive data in binary format.
BufferedStream	A temporary storage for a stream of bytes.
Directory	Helps in manipulating a directory structure.
DirectoryInfo	Used for performing operations on directories.
DriveInfo	Provides information for the drives.
File	Helps in manipulating files.
FileInfo	Used for performing operations on files.
FileStream	Used to read from and write to any location in a file.
MemoryStream	Used for random access of streamed data stored in memory.
Path	Performs operations on path information.
StreamReader	Used for reading characters from a byte stream.
StreamWriter	Is used for writing characters to a stream.
StringReader	Is used for reading from a string buffer.
StringWriter	Is used for writing into a string buffer.

১. ফাইলস্ট্রিম ক্লাস: FileStream Class

এ ক্লাসের মাধ্যমে আমরা নির্দিষ্ট ফাইল খুলে পড়া, লেখা ও বন্ধ করার মতো কাজ করতে পারি। একটি উদাহরণ থেকে এর ব্যবহার স্পষ্ট হবে।

এটি একটি অতি সাধারণ উদাহরণ। ফাইল নিয়ে কাজ করার একাধিক শক্তিশঅলী ক্লাস আছে। পরের পৃষ্ঠা থেকে বেশ ক'টি উদাহরণ শিক্ষার্থীদের সুবিধার্থে উপস্থাপন করেছি। খুব ভালো করে ওগুলো স্টাডি করবেন। ফাইল নিয়ে কাজ করা প্রোগ্রামিংয়ের সঙ্গে অঙ্গাঙ্গিভাবে জড়িত। সুতরাং একে জানা একান্ত জরুরী।

```
Imports System.IO
Module fileProg
  Sub Main()
    Dim f1 As FileStream = New FileStream("sample.txt", _
         FileMode.OpenOrCreate, FileAccess.ReadWrite)
    Dim i As Integer
    For i = 0 To 20
       f1.WriteByte(CByte(i))
    Next i
    f1.Position = 0
    For i = 0 To 20
       Console.Write("{0} ", f1.ReadByte())
    Next i
    f1.Close()
    Console.ReadKey()
  End Sub
End Module
```

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

ক. টেস্টট্ ফাইল থেকে পড়া ও লেখা: এ জন্য মাত্র ক'টি কোড লাইনই যথেষ্ট। যেমন:

```
Private Sub ReadTxtFile
    Dim R as New IO.StreamReader("") 'put the full file path between the quote marks
    txtText.Text = R.ReadToEnd
    R.Close()
End Sub
```

উক্ত কোড প্যাথ দ্বারা চিহ্নিত টেক্সট্ ফাইল খুলে টেক্সট্ বাক্সে দেখাবে। স্ট্রিমরিডার ক্লাসকে ফাইল পাঠের পর বন্ধ করতে হবে। সুতরাং তৃতীয় লাইনটি এখানে জরুরী।

খ. টেস্টট্ ফাইল থেকে পড়া, এডিট করা ও সেইভ করা: এ কাজগুলোও তেমন বেশি কঠিন নয়। আপনার সলুশনে যুক্ত করুন একটি OpenFileDialog ও একটি SaveFileDialog অবজেক্ট। এরপর নিচের কোডটুকু লিখুন। যথাক্রমে অপেন ও ক্লোজ বোতাম থেকে সাবগুলো কল করুন।

```
imports System.IO 'এটা উপরে ডিক্লায়ার হবে।
Private Sub ReadWriteTxtFile
    Dim FileReader as StreamReader
    Dim Results as DialogReasult
    Resutls = OpenFileDialog1.ShowDialog
    If Results = DialogReasult.OK then
        FileReader = New StreamReader(OpenFileDialog1.FileName)
        txtText1.Text = FileReader.ReadToEnd()
        FileReader.Close()
    End If
End Sub

Private Sub ReadWriteTxtFile
    Dim FileWriter as StreamWriter
    Dim Results as DialogReasult
    Resutls = SaveFileDialog1.ShowDialog
    If Results = DialogReasult.OK then
        FileWriter = New StreamWriter(SaveFileDialog1.FileName, False)
        FileWriter.Write(txtText.Text)
        FileWriter.Close()
    End If
End Sub
```

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

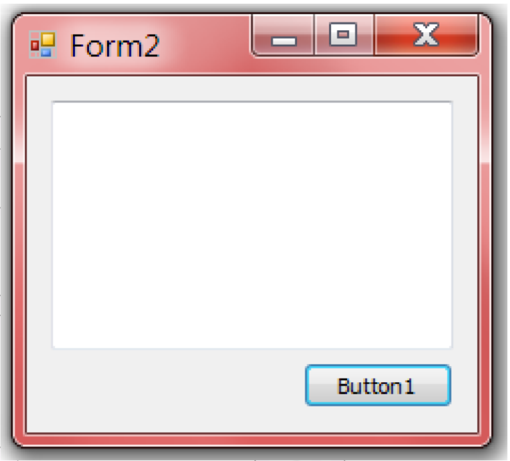
ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

২২. ফর্মে কন্ট্রোল ডক ও নঙ্গর করা: Docking and Anchoring Controls on the Form

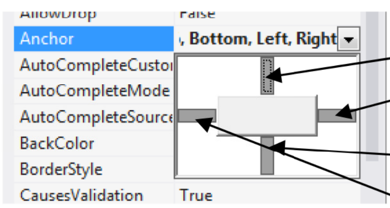
ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিংয়ের ওপর রচিত এ গ্রন্থের শেষ পর্যায়ে এসে গেছি আমরা! বাস্তবে জানার তো শেষ নেই। কিন্তু এক পর্যায়ে তো গ্রন্থের ইতি টানতেই হবে- তাই না? আমি আশাকরি শিক্ষার্থীরা এ গ্রন্থ পাঠ করে ও এতে সন্নিবেশিত বিভিন্ন উদাহরণ এ্যাপলিকেশন থেকে উপকৃত হয়েছেন। ভিজুয়্যাল বেসিক.নেট একটি শক্তিশালী উইন্ডোজ প্রোগ্রামিং ভাষা। শিক্ষার্থীদের প্রতি আমার উপদেশ হচ্ছে, ইন্টারনেট থাকলে [আর আজকাল কার ইনন্টারনেট নেই!] ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং শেখার অসংখ্য ওয়েভসাইট থেকে উপকৃত হতে পারেন। আপনি যা কিছুই করতে চান না কেনো, সেসব বিষয়ের ওপর উদাহরণ কোড ও প্রজেক্ট অবশ্যই খুঁজে পাবেন। সময় সময় কিছুটা অতিরিক্ত ধৈর্যশীল হতে হবে- এই যা।

যাক, আমরা একটি গুরুত্বপূর্ণ বিষয়ের ওপর কিঞ্চিৎ আলোচনা করে এ রচনার ইতি টানছি। বিষয়টি মূলত ইউজার ইনন্টারফেইস ডিজাইন সম্পর্কিত। ব্যবহারকারী আপনার সৃষ্ট এ্যাপলিকেশনের জানালাকে প্রয়োজনে বড়- ছোট করতে পারেন। জানালায় থাকতে পারে টেক্সট বাক্স, বোতাম ইত্যাদি। সুতরাং ফর্ম রিসাইজ হলে এগুলোও যাতে তার জায়গায় থেকে যায় সে ব্যবস্থা জরুরী। ডিজাইনকালে দু'টি প্রোপার্টি সেটিং দ্বারা তা হতে পারে। এগুলো হলো: ১. ডকিং ও অ্যাঙ্কোরিং। ডকিংয়ের কাজ, যে বস্তর ওপর সে বসে আছে- যেমন ফর্ম, তার সাথে নিজের সাইজও বাড়ানো ও কমানো।

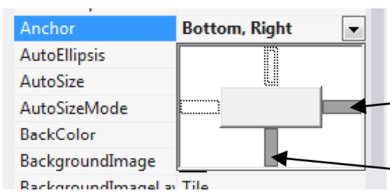
একটি কোডহীন উদাহরণ দ্বারা বিষয়টি বুঝিয়ে দিচ্ছি। ডানের চিত্রে আমরা একই ফর্ম রানকালে বড়ো ও ছোট সাইজে দেখতে পাচ্ছি। ফর্মের মধ্যে দু'টি অবজেক্ট আছে: বোতাম ও টেক্সট বাক্স। ব্যবহারকারী ইচ্ছে মতো ফর্মের সাইজ কমাতে বা বাড়াতে পারবেন। সাথে সাথে বোতামটি নিজের সাইজ অপরিবর্তিত রেখে স্বস্থানে থাকবে। অপরদিকে টেক্সট বাক্সের সাইজও প্রপোরশন্যালি বাড়বে কমবে। এই দু'টি ক্রিয়া নিশ্চিত করতে যেয়ে আমাদেরকে ডিজাইন টাইমে ফর্মের ও বোতামের Anchor নামক প্রোপার্টি সঠিকভাবে সেট করতে হবে।



টেক্সট বাক্সের উপর কোথাও ক্লিক করুন। প্রেপার্টিতে যেয়ে Anchor -এর ডানের এ্যারো চয়েজ করুন। ছোট্ট ৪টি চতুর্ভুজের রঙ গ্রে করুন একটা একটা করে ক্লিক করে [দেখুন নিচের প্রথম চিত্রটি]। এরপর বোতামে ক্লিক করে Anchor প্রোপার্টিতে যেয়ে এর শুধুমাত্র ডানের ও নিচের চতুর্ভুজ দু'টোর রঙ গ্রে করে দিন ক্লিক করে [দেখুন নিচের দ্বিতীয় চিত্রটি]। ব্যস! ফর্মের সাথে সঠিকভাবে টেক্সট বাক্স রিসাইজ হবে ও বোতাম তার জায়গায় থাকবে।



টেক্সট বাক্সের Anchor -এর চারটি ছোট্ট বাক্সই গ্রে হবে।



বোতামের ক্ষেত্রে গ্রে থাকবে শুধুমাত্র ডানেরটি ও নিচেরটি।

